COMPUTER NETWORK RESEARCH

Leonard Kleinrock

California University

Prepared for:

Advanced Research Projects Agency

31 December 1973

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD/A-004 167
4. TITLE (and Subtitle) Computer Network Research		5. TYPE OF REPORT & PERIOD COVERED Semi-Annual Technical Report July-December, 1973 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Leonard Kleinrock		8. CONTRACT OR GRANT NUMBER(s) DAHC 15-73-C-0368
9. PERFORMING ORGANIZATION NAME AND ADDRESS School of Engineering and Applied Science University of California Los Angeles, California 90024		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 2496
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209		12. REPORT DATE December 1973 13. NUMBER OF PAGES 312
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Sponsored by

ADVANCED RESEARCH PROJECTS AGENCY

COMPUTER NETWORK RESEARCH

SEMIANNUAL TECHNICAL REPORT

December 31, 1973

ARPA Contract DAHC-15-73-C-0368*

ARPA Order No. 2496
Program Code No. 3P10

Principal Investigator: Leonard Kleinrock

Co-Principal Investigators: Gerald Estrin
Michel Melkanoff
Richard R. Muntz
Gerald Popek

Computer Science Department
School of Engineering and Applied Science
University of California, Los Angeles

COMPUTER NETWORK RESEARCH

Advanced Research Projects Agency
Semiannual Technical Report

December 31, 1973

This Semiannual Technical Report describes the result of one of our major activities during the period July 1, 1973 through December 31, 1973. It concerns the analysis and control of a satellite channel used in a packet switching mode. This study constitutes only one of many undertakings with the support of ARPA Contract DAHC 15-73-C-0368 during this period. It represents one coherent and significant piece of research to which we devote this report.

The main body of this report contains a reproduction of Simon Lam's Ph.D. thesis (supervised by Leonard Kleinrock) entitled, "Packet Switching in a Multi-access Broadcast Channel with Application to Satellite Communication in a Computer Network". He completed this dissertation in March, 1974 and it was published as a report in the Computer Science Department, UCLA-ENG-7429 in April 1974.

The research is concerned with the shared use of a broadband satellite channel for computer communications. The objective is to allow a large number of independent earth stations to simultaneously share the entire capacity of the given channel in a random fashion. This multi-access uplink is subject to conflicts in that more than one user may attempt to send his fixed length packet at the same time thereby causing a destructive interference among such conflicting packets. The satellite is realistically treated as a pure transponder which broadcasts back to all earth stations within its shadow in exact replica of that which it receives (a complete packet or an interfered with transmission).

Since each earth station hears the same transmission from the satellite then any station which transmits its packet will, after a round trip time delay of approximately one quarter of a second, also hear its own transmission and will be able to determine if a destructive conflict occurred. Based on this observation the study then evaluates the performance under a particular random access mode referred to as slotted aloha. It is then shown that these channels stable with very little loss in throughput. The main contributions then involve the stability to put delay trade-offs for unstable channels and the dynamic control and estimation procedures for rendering these channels stable.

This study fits into the use of satellite packet-switching for the satellite IMPs (SIMPS) currently being considered by ARPA for their intercontinental packet-switching satellite network experiments.

LIST OF PUBLICATIONS

Chu, W. W., "Dynamic Buffer Management for Computer Communications," presented at the ACM Third Data Communication Symposium, Tampa, Florida, November 1973.

Kleinrock, L., "Challenging Problems in the Design of Computer-Communication Networks," Proceedings of the XX International Meeting of the Institute of Management Sciences (TIMS XX), Tel Aviv, Israel, June 23-July 8, 1973.

Kleinrock, L., "Analytical Techniques for Computer-Communications Networks," International Seminar on "Computers and Communications," University of Newcastle-on-Tyne, England, September 4-7, 1973.

Kleinrock, L., "Computer Networks: Issues and Challenges," Session on Networking: Applications and Their Network Requirements," Computer Telecommunications Conference, organized by the International Institute for the Management of Technology (IIMT), October 1-4, 1973, Milan, Italy.

Kleinrock, L., "Scheduling, Queueing and Delays in Time-Shared Systems and Computer Networks," in Computer-Communication Networks, N. Abramson and F. Kuo, (eds.), Prentice-Hall, Englewood Cliffs, N.J., 1973, pp. 95-141.

Kleinrock, L. and S. S. Lam, "Dynamics of the ALOHA Channel," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ARPANET Satellite System Note 50, Network INformation Center #18455, August 1973.

Kleinrock, L. and S.S. Lam, "On Stability of Packet Switching in a Random Multi-Access Broadcast Channel," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974 Proceedings of the Special Subconference on Computer Nets, 1974, also ARPANET Satellite System Note 52, Network Information Center #19934, November 1973.

Kleinrock, L. and F. Tobagi, "Throughput-delay Tradeoffs for Reservation Access Modes in Packet-Radio Systems," Packet Radio Temporary Note No. 68, UCLA ARPA Network Measurement Center, Computer Science Department, July 2, 1973.

Kleinrock, L. and F. Tobagi, "Performance of CArrier Sense with Hidden Terminals," Packet Radio Temporary Note No. 75, UCLA ARPA Network Measurement Center, Computer Science Department, October 10, 1973.

Lam, S. S., "Some Satellite Simulation Results," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ARPANET Satellite System Note 48, Network Information Center #17655, July 1973.

Muntz, R. R. and F. Baskett "Open, Closed and Mixed Networks of Queues with Different Classes of Customers," accepted for publication in JACM.

Muntz, R. R. and H. Opderbeck, "Stack Algorithms for Two-Level Directly Addressable Paged Memories," accepted for publication in SIAM Journal on Computing.

Naylor, W. E., "On Message Delay Over World-Wide Channels," Network Working Group Note 14, Network Information Center #19013, July 1973.

Naylor, W. E., "Real-Time Transmission in a Packet Switched Network," Network Working Group Note 15, Network Information Center #19014, September 1973.

Popek, G., "Notes on the Value of Security and Virtual Machines," ARPA System Security Project Internal Memorandum #4, University of California, Los Angeles, July 1973.

Popek, G., "Correctness in Access Control, "Proceedings of the Association for Computing Machinery National Conference, Atlanta, Georgia, pp. 236-241, August 1973.

Popek, G. and C. Kline, "UCLA Security Project," Internal Memorandum #5, University of California, Los Angeles, August 1973.

UCLA-ENG-7429
April 1974

PACKET SWITCHING IN A MULTI-ACCESS BROADCAST CHANNEL WITH APPLICATION TO SATELLITE COMMUNICATION IN A COMPUTER NETWORK

by

Simon S. Lam

School of Engineering and Applied Science
University of California
Los Angeles

ABSTRACT

This report considers a packet switching technique applicable to packet communication using a satellite or ground radio channel. The objective of this research is to develop analytic models for the evaluation and optimization of the system performance in terms of stability, throughput and delay.

Advantages of packet switched satellite and ground radio systems over conventional wire communications for large computer-communication networks are discussed. The emphasis of this research is on a high-speed channel shared by a large population of "small" users. The channel behavior is typical of "contention" systems in which the throughput vanishes to zero as the load on the system increases. This phenomenon is called channel saturation. The channel may go into saturation as a result of (a) time fluctuations, and (b) stochastic fluctuations in the channel input. The channel response to time varying inputs is first studied using a deterministic approximation analysis. The effect of (b) is then studied through probabilistic models. In this case, contributions of this research may be classified into three categories:

(1) a coherent theory of channel behavior in which the key result is the characterization of stable and unstable channels

(2) evaluation of channel performance such as equilibrium throughput-delay tradeoffs for stable channels and stability-throughput-delay tradeoffs for unstable channels

(3) dynamic channel control and estimation procedures for optimal control of unstable channels.

This study has several implications. First, a coherent theory of channel behavior has been developed, system design variables have been identified and operational strategies for the optimization of channel performance have been evaluated. These results suggest a system design methodology. Second, the techniques employed in characterizing the stability behavior and evaluating dynamic channel control schemes may profitably be applied to probabilistic models of other contention systems.

CONTENTS

CONTENTS (continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In the design of a computer-communication network, two problems may be identified. One is to provide long haul communications among geographically scattered computers and resources. The other is to provide local distribution of the network computing power, communication power and resources to populations of users.

An abstract model of a computer-communication network is depicted in Fig. 1-1. The first problem mentioned above corresponds to the design of the communication subnet in the figure for computer-

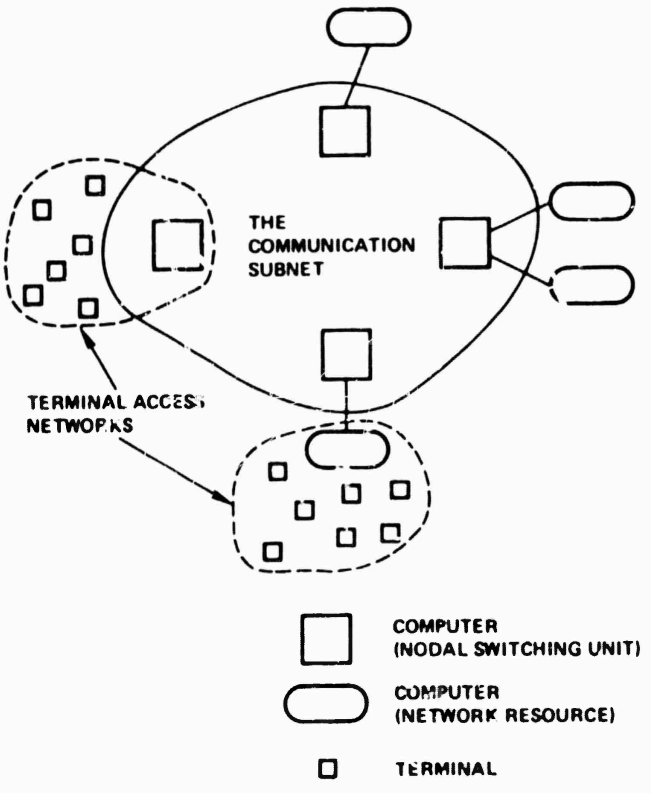


Figure 1-1. An Abstract Model for a Computer-Communication Network.

computer communications. The second problem corresponds to the design of the terminal access networks for terminal-computer communications. Two kinds of computers are distinguished in the model: (1) autonomous computer systems which constitute resources to be shared over the network, and (2) special purpose processors dedicated to network communication functions and acting as nodal switching units for data flow. (These nodal switching units will be referred to as the nodes of a communication subnet.)

The above abstract model description of a computer-communication network is consistent with the design philosophy of the ARPA (Advanced Research Projects Agency) Network [BUTT 74, CARR 70, CROC 72, FRAN 70, FRAN 72A, HEAR 70, KLEI 70, ORNS 72, ROBE 70, ROBE 72A].

In this dissertation, a packet switching technique based upon the random access concept of the ALOHA System [ABRA 70] will be studied in detail. This technique enables efficient sharing of a communication channel by a large population of users, each with a bursty input source (large ratio between the peak and average data rate). This packet switching technique may be applied to the use of satellite and ground radio channels for computer-computer and terminal-computer communications respectively. The multi-access broadcast capabilities of these channels render them attractive solutions to (1) large communication subnets with nodes over wide geographically distributed areas, and (2) large terminal access networks with potentially mobile terminals.

1.1 Present Computer-Communication Schemes

The simplest solution to providing communication between two points is to assign a dedicated channel for their use. This method

is expensive in computer communications especially over long distances. Measurement studies [JACK 69, FUCH 70] conducted on time-sharing systems indicate that both computer and terminal data streams are bursty. That is, the peak data rate is much larger than the average data rate. (The ratio between them may be as high as 2000 to 1 [ABRA 73].) Consequently, if a high-speed point-to-point channel is used, the channel utilization is low since the channel is idle most of the time. On the other hand, if a low-speed channel is used, the transmission delay is large.

The above dilemma is caused by channel users imposing bursty random demands on their communication channels. By the law of large numbers in probability theory [FELL 68], the total demand at any instant of a large population of users is, with a high probability, approximately equal to the sum of their average demands. Thus, if a channel is dynamically shared in some fashion among many users, the required channel capacity may be much less than the unshared case of dedicated channels. This concept is known as statistical load averaging and has been applied in many computer-communication schemes to various degrees of success. These schemes include: polling systems [MART 72], loop systems [HAYE 71, PIER 71], Asynchronous Time Division Multiplexing (ATDM) [CHU 69], the random access scheme in the ALOHA System, and the store-and-forward packet switching concepts [BARA 64, KLEI 64, DAVI 68] implemented in the ARPA Network.

For almost a century, circuit switching dominated the design of communication networks. Only with the speed and cost of modern

computers did packet communication become competitive. It was not until 1970 that the computer (switching) cost dropped below the communication (bandwidth) cost [ROBE 74]. This also marked the first appearance of packet switched computer-communication networks.

In a circuit switched network, a complete path of communication links must be established between two parties before they can communicate. The path (of links) is allocated for as long as the two parties want. In a store-and-forward packet switched network, the communication is broken into convenient size packets of information with addresses of source and destination attached to each packet. Packets are individually routed through the network to their destinations "hopping" from one node to another. In this case, the communication links are not allocated into paths for specific source-destination pairs of nodes; instead, each link is statistically shared by many nodes. The large savings possible from fuller utilization of the communication links justify the extra computer switching cost.

1.2 Satellite and Radio Communications in Large Networks

We are currently facing a booming demand for computer networks. For example, a survey for 17 European nations entitled "Eurodata--A Market Study on Data Communications in Europe, 1972-1985" estimates that data communication volume in those countries will soar twelvefold in the next dozen years. The total number of terminals was 79,600 in 1972; it will rise to 235,600 by 1976 and to 815,000 by 1985 [WRIG 73]. The feasibility of packet switched networks with up to 1000 nodes and tens of thousands of terminals is being investigated [NAC 73, FRAN 73].

These numbers are at least an order of magnitude larger than any other system design attempted. Extension of current computer-communication techniques to networks of such magnitude cannot be easily done. For instance, the adaptive routing techniques currently implemented in the ARPA network cannot be directly utilized in a very large network because of excessive IMP processing time, memory requirements and traffic overhead [NAC 73]. The system overhead in conventional polling schemes is directly proportional to the number of terminals sharing the communication channel; such schemes are thus not appropriate for a large number of terminals.

To design cost-effective computer-communication networks for the future, new techniques are needed which are capable of providing efficient high-speed computer-computer and terminal-computer communications in a large network environment.

The application of packet switching techniques to radio communication (both satellite and ground radio channels) provides a solution.

Radio is a multi-access broadcast medium. A signal generated by a radio transmitter may be received over a wide area by any number of receivers. (This is the broadcast capability.) Furthermore, any number of users may transmit signals over the same channel.* (This is the multi-access capability.) Hence, a single ground radio channel provides a completely connected network topology for a large number

*However, if two signals (packet transmissions) at the same carrier frequency overlap in time at a radio receiver, we assume that neither is received correctly.

of nodes within line of sight of each other. On the other hand, a satellite transponder in geosynchronous orbit above the earth acts as a radio repeater. Any number of earth stations may transmit signals up to the satellite at one carrier frequency (the multi-access channel). Any signal received by the satellite transponder is beamed back to earth at another frequency (the broadcast channel). This broadcasted signal may be received by all earth stations covered by the transponder beam. Thus, a satellite channel (consisting of both carrier frequencies) provides a completely connected network topology for all earth stations covered by the transponder beam (see Fig. 1-2).

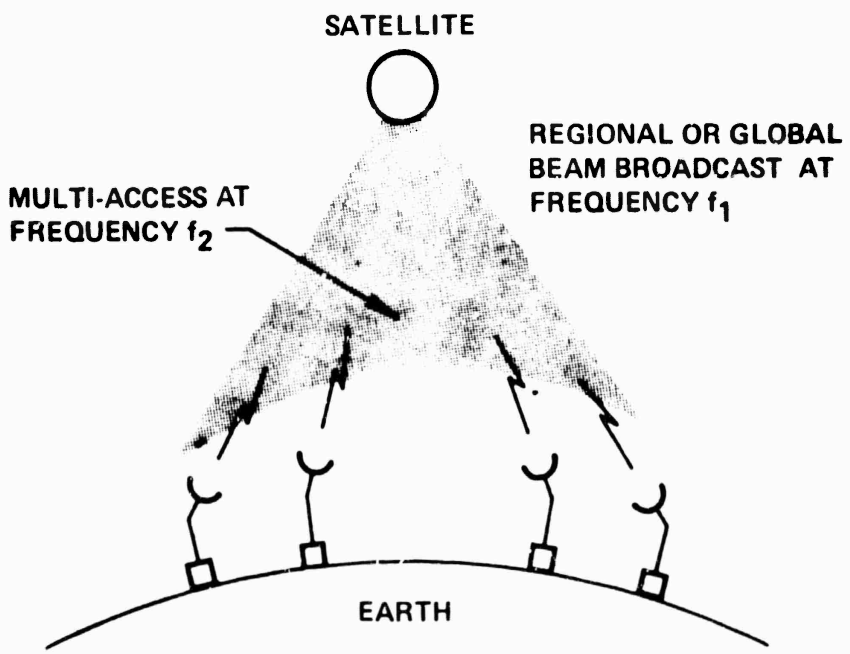


Figure 1-2. Packet Switch in the Sky.

The provision of a completely connected network topology by a satellite or radio channel eliminates complex topological design and routing problems in large networks [FRAN 72B, GERL 73]. Moreover, the use of packet switching techniques enables a large population of users to statistically average their total load at the high-speed multi-access channel. Each user also transmits data at the (wideband) data rate of the channel. Thus, both high channel utilization and small packet delays are possible through the use of appropriate packet switching techniques. We shall elaborate upon the advantages of packet switched radio communication systems in the next chapter.

We give here a description of the ALOHA System which is one of the first packet radio communication systems.

The ALOHA System is an experimental terminal access network at the University of Hawaii [ABRA 70, KUO 73]. Two 24 KBPS radio channels are used. One of the two channels is used by all remote terminals for transmitting data into the central computer. (This is a multi-access channel.) The other channel is used for transmitting data out of the central computer to the remote terminals. (This is a broadcast channel.) The transmission of data from the central computer to the terminals is relatively simple since the central computer can schedule its own use of the broadcast channel. The multi-access channel, however, uses the following radically new random access packet switching technique. (This scheme will be referred to as pure ALOHA.) Each terminal transmits data to the central computer over the same 24 KBPS channel in 30 msec. bursts

(packets) in a completely unsynchronized manner. A transmitted packet can be received incorrectly as a result of two types of errors: (1) random noise errors, and (2) errors caused by interference (at the radio receiver of the central computer) with a packet transmitted by another terminal. If and only if a packet is received with no error, it is acknowledged by the central computer. After transmitting a packet, a terminal waits a given amount of time (time-out interval) for an acknowledgment; if none is received, the packet is retransmitted. This process is repeated until successful transmission and acknowledgment occur or until the process is stopped by the terminal. It was estimated that the ALOHA System could theoretically support more than 300 active terminals [ABRA 70].

There is currently an immense worldwide interest in the development of satellite communications systems. In addition to the worldwide INTELSAT system [PUEN 71], there are currently in operation two domestic satellite systems: Anik in Canada [GRAY 74] and Molniya in the U.S.S.R. With the advent of domestic satellite systems in the United States [CACC 74], various satellite computer-communication systems based upon the packet radio communication concept of the ALOHA system have been proposed [ABRA 73, CROW 73, KLEI 73A, ROBE 73]. In particular, Abramson suggested that a single transponder in a domestic satellite system could easily provide 10 MBPS for a public packet switched service with 100 earth stations over the U. S.; each earth station has an average data rate of 15 KBPS and a maximum transmission rate of 10 MBPS [ABRA 73]. Dunn and Eric gave a comparison

of illustrative costs for some of the above proposed packet switched satellite systems assuming the use of small earth stations for 100 nodes serving the 40 largest metropolitan areas in the U. S. [DUNN 74]. In a recent application to the FCC for approval of a public packet switched network, a 1.5 MbPS satellite channel was included in the proposed network configuration based on land lines [TELE 73].

1.3 Packet Switching Techniques

Consider a radio communication system such as the satellite system depicted in Fig. 1-2 or the ALOHA System. In each case, there is a broadcast channel for point-to-multipoint communication and a multi-access channel shared by a large number of users. Each user is assumed to have a small average data rate relative to the channel transmission rate, but each transmits packets of data at the channel transmission rate. (In other words, the users have bursty input sources.)

Since the broadcast channel is used by a single transmitter, no transmission conflict will arise. All nodes covered by the radio broadcast can receive on the same frequency, picking out packets addressed to themselves and discarding packets addressed to others.

The problem we are faced with is how to effect time-sharing of the multi-access channel among all users in a fashion which produces an acceptable level of performance. As soon as we introduce the notion of sharing in a packet switching mode, we must be prepared to resolve conflicts which arise when simultaneous demands are placed upon the channel. There are two obvious solutions to this problem:

the first is to form a queue of conflicting demands and serve them in some order; the second is to "lose" any demands which are made while the channel is in use. The former approach is taken in ATDM and in a store-and-forward network assuming that storage may be provided economically at the point of conflict. The latter approach is adopted in the ALOHA System random access scheme; in this system, in fact, all simultaneous demands made on the radio channel are lost.

Let us define channel throughput rate S_{out} to be the average number of correctly received packet transmissions per packet transmission time (assuming stationary conditions). We also define channel capacity S_{max} to be the maximum possible channel throughput rate.

The channel capacity of a pure ALOHA multi-access channel was estimated to be $\frac{1}{2e} \cong 18\%$ for a fixed packet size [ABRA 70]. Under similar assumptions, Gaarder showed that a pure ALOHA channel with a fixed packet size is always superior (in terms of channel capacity) to one with different packet sizes [GAAR 72].

Since various propagation delays are involved in a geographically distributed radio communication system, let us define a global reference time called channel time. The channel time will be assumed to be the satellite transponder time in a satellite system and to be the central computer time in a terminal access network. Note that if two or more packet transmissions overlap in time at the radio receiver (of the satellite transponder or the central computer), none is received correctly. This event will be referred to as a channel collision.

Roberts suggested that the channel time may be slotted by requiring all channel users to synchronize the leading edge of each packet transmission to coincide with an imaginary channel time slot boundary. The duration of a channel time slot is equal to a packet transmission time. The resulting scheme will be referred to as "slotted ALOHA random access" or "slotted ALOHA." (In Fig. 1-3, we show packet transmissions and retransmissions in a slotted ALOHA system consisting of four users.) The channel capacity of a slotted ALOHA channel was estimated* to be $\frac{1}{e} \cong 36\%$ [ROBE 72B].

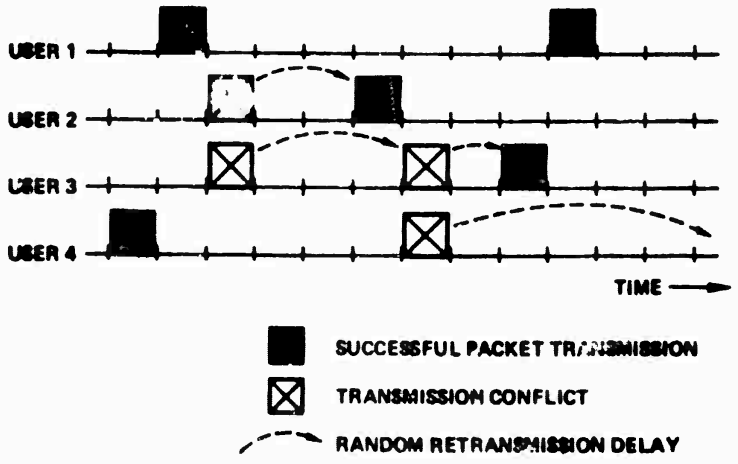


Figure 1-3. Slotted ALOHA Random Access.

Several variants of the ALOHA random access technique have been proposed for ground radio systems. One technique is known as FM capture [ROBE 72B]. In the event of a channel collision, the

*A derivation of this result is given in Chapter 3.

strongest signal (packet transmission) may still be received correctly by a good FM receiver. As a result, the ALOHA random access channel capacity may be larger than the 36% limitation. Another scheme is currently being investigated for ground radio packet switching systems in which the maximum propagation delay is small compared to a packet transmission time (say, less than 0.1). In such systems, the users may "listen before transmitting" in order to determine if the channel is in use by some other user; such systems are referred to as "carrier sense" systems. In these systems, a channel capacity much larger than 36% is possible [KLEI 74C].

Several reservation schemes based upon the slotted ALOHA random access technique have been proposed for satellite packet switching systems. In a satellite channel, the round-trip propagation delay is approximately a quarter of a second and is in the order of many channel time slots. In one reservation system [ROBE 73], the satellite channel is dynamically partitioned into a slotted ALOHA channel for broadcasting reservation requests and a scheduled channel for transmitting multi-packet blocks of data. The minimum delay in this system is at least twice the round-trip propagation delay (half a second). Thus, this sytem is preferable if a significant fraction of the channel input source consists of multi-packet messages and if the average message delay is the relevant measure of channel performance. Two "reservation-ALOHA" schemes have also been proposed [BIND 72, CROW 73]. These schemes may be used if there is only a small number of channel users (say, in the order of the number of

slots in a round-trip propagation time), and if the channel input source has constant as well as random components.

1.4 Summary of Results

We examined several radio communication packet switching schemes in the last section. Some of these schemes (FM capture, carrier sense, reservation-ALOHA) are variants of the slotted ALOHA random access concept; some others (e.g., Robert's reservation system) are dependent upon the slotted ALOHA random access technique.

The basic goal of this dissertation is to develop analytic models with which we can predict and optimize the stability-throughput-delay performance of a multi-access channel using the slotted ALOHA random access technique. The analytic models, despite their limitations (due to various mathematical assumptions), suggest a system design methodology and operational strategies for packet switching random access systems. Our emphasis is on a large population of users with bursty input sources; each user has an average data rate which is small relative to the channel transmission rate.

It has been realized that in a slotted ALOHA random access channel, channel "saturation" may occur as a result of time fluctuations in the channel input or innerent channel instability [KLEI 73A, KLEI 73B, KLEI 74A, LAM 73, METC 73A, RETT 72]. However, existing results on the channel capacity [ABRA 73] and throughput-delay tradeoff [KLEI 73A] have all assumed steady-state conditions. A channel control strategy derived from a steady-state analysis has been proposed which may prevent channel saturation [METC 73A].

Considering the state of the research, only fragmented results are available on the performance (channel capacity, delay, dynamic behavior and stability) of the slotted ALOHA random access channel. Little attention has been paid to the problem of dynamic channel control. In this dissertation, we attempt to give a coherent theory of channel behavior and to develop techniques to optimize the system design and dynamically control the channel performance.

In Chapter 2, we summarize various advantages of satellite and radio communications over conventional wire communications. Satellite channel characteristics and cost trends are examined. Abstract models are then given for the random access channel and channel users to be considered in the dissertation.

In Chapter 3, an analytic model is developed to predict the equilibrium throughput-delay tradeoff. The minimum throughput-delay performance envelope and the corresponding optimal retransmission delays are characterized. These results are generalized to a model which includes a "large" user in the channel user population.* In this case, significant improvements in the channel throughput-delay performance are possible. A channel throughput rate equal to one may be achieved. A continuum of throughput-delay performance envelopes are presented. Abramson's result [ABRA 73] on channel capacity will also be given. Simulation results have been obtained which agree very well with analytic results. However, the assumption of channel

*This situation arises when, for example, in a terminal access packet radio system, a single radio channel is used for both terminal-to-computer (multi-access) and computer-to-terminal (broadcast) communications [GITM 74].

equilibrium may be valid only for finite time periods beyond which the channel goes into saturation.

In Chapter 4, the complexity of an exact mathematical analysis of channel dynamics is illustrated. This serves to motivate our use of approximations. The channel traffic (packet transmissions and retransmissions in a channel slot) is shown to be Poisson distributed in the limit of an infinite average retransmission delay and under the "weak independence assumption." A difference equation is derived which gives a deterministic approximation of the dynamic behavior of the channel subject to time varying inputs.

In Chapter 5, stable and unstable channels are characterized and a stability definition is given. For stable channels, previous equilibrium throughput-delay results given in Chapter 3 are actually valid and achievable over an infinite time horizon. For unstable channels, the degree of instability is quantified by the definition of the stability measure FET. An efficient algorithm is developed for the calculation of FET. Unstable channels, in general, are characterized by a large population of users. The "stability" (i.e., FET) of an unstable channel may be improved by reducing the channel input rate or increasing the average packet delay The appropriate channel performance measure for unstable channels is the stability-throughput-delay tradeoff. Some stability-throughput-delay tradeoff curves are presented.

Under the assumption that channel users have bursty input sources with low data rates (relative to the channel speed), stable channels are characterized by a relatively small population of users

and thus, a small throughput rate. To obtain a high channel throughput rate, dynamic channel control is necessary to convert unstable channels into stable channels. In Chapter 6, Markov decision theory is used to formulate three dynamic channel control procedures (ICP, RCP, IRCP). It is shown that optimal stationary policies exist. Furthermore, a theorem is proved that the same stationary control policy will maximize the stationary channel throughput rate and minimize the average packet delay simultaneously. An efficient computational algorithm (POLITE) is developed which utilizes Howard's policy-iteration method [HOWA 71] and is capable of solving for an optimal stationary policy in a small number of computational steps. Numerical results indicate that optimal control policies are of the control limit type, but a rigorous mathematical proof remains an open problem. Throughput-delay tradeoffs given by optimal control policies are presented. These throughput-delay results are very close to the optimum performance envelope in Chapter 3 and are achievable over an infinite time horizon for (originally) unstable channels. Since in a practical system the exact channel state is not known but must be estimated, some channel control-estimation (CONTEST) algorithms based upon the dynamic control procedures are proposed. A heuristic control algorithm is also suggested. Simulations indicate that for a channel throughput rate up to 0.32, throughput-delay results close to the optimum channel performance are achievable through application of the CONTEST algorithms.

In Chapter 7, multi-packet messages are considered. An approximate formula for the average message delay is derived. Roberts' reservation system and two reservation-ALOHA schemes are surveyed.

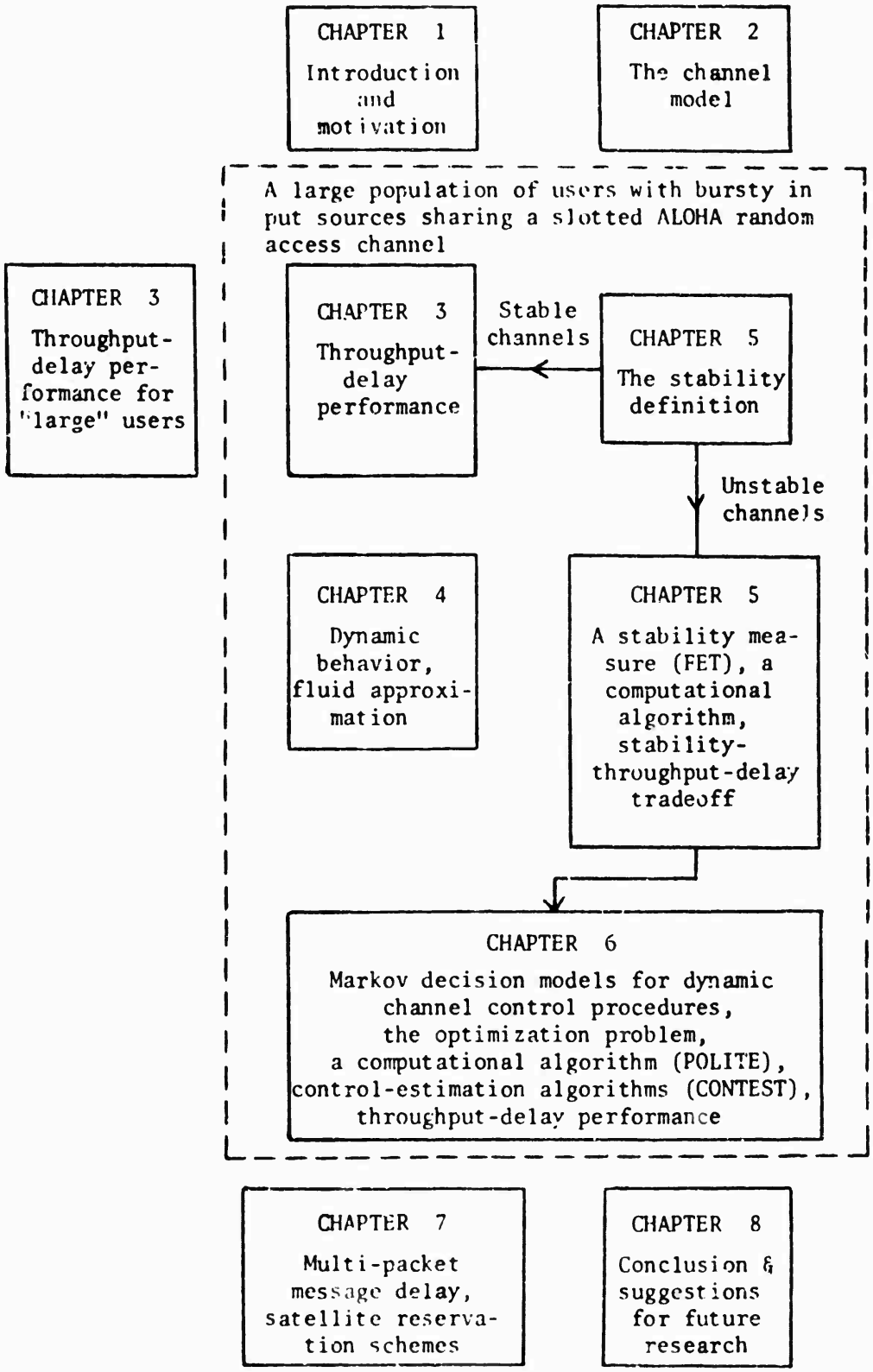


Fig. 1-4 Summary of results in this dissertation.

In Chapter 8, we give some concluding remarks and suggest topics of future research interests.

The above summary of results in this dissertation is depicted in Fig. 1-4.

This research was motivated by the research and development activities of the ARPANET Satellite System intended to incorporate satellite packet communication into the existing ARPA Network [ABRA 72, BUTT 74]. Consequently, the use of a satellite channel is considered in numerical examples throughout this dissertation. A satellite channel is characterized by a large channel propagation delay which will be reflected in all our numerical results. However, the models and methodology developed in this dissertation are applicable to ground radio systems. In fact, before small satellite earth stations become a reality (economically), the assumption of a large population of channel users is more appropriate in a ground radio environment. We also note that application of the random access techniques considered here is not limited to satellite and radio multi-access broadcast channels. They can, for example, also be applied to terminal access networks with multi-drop lines [HAYE 72].

In summary, the major contributions of this research are:

(1) The characterization and performance evaluation of stable and unstable channels--for stable channels, techniques are developed to solve for the optimum throughput-delay performance envelope. For unstable channels, the degree of channel instability is

quantified by the definition of the stability measure FET. An efficient algorithm has been developed to calculate FET. The channel stability-throughput-delay performance is shown.

(2) Dynamic channel control procedures which prevent channel saturation in an unstable channel to give better channel utilization--Markov decision models are developed for various dynamic control procedures. Optimal stationary control policies are shown to exist which will maximize the stationary channel throughput rate and minimize the average packet delay simultaneously. An efficient algorithm (POLITE) based upon the policy-iteration method finds an optimal stationary policy in a small number of computational steps. Control-estimation (CONTEST) algorithms are proposed for practical implementation of the above control procedures. Truly stable channel throughput-delay performance close to the optimum performance envelope is achievable using the dynamic control procedures.

In conclusion, despite model limitations as a result of various assumptions for mathematical convenience, we feel that the results and methodology presented in this dissertation are valuable and will lead to sound design procedures and operational strategies for packet communication systems using radio and satellite channels in a large network environment.

CHAPTER 2

THE CHANNEL MODEL

The multi-access packet switching techniques introduced in the last chapter may be applied to wire communications as well as radio communications (both satellite and ground radio) [HAYE 72]. For example, a multi-drop line can be used in either the multi-access or broadcast mode; also, a loop system can be used as a multi-access broadcast system. However, as we mentioned before, we are interested in the use of radio packet communication for large populations of users over wide areas. With this in mind, we discuss below some advantages of radio communications over conventional wire communications. Since this research is motivated by the ongoing research and development of the ARPANET Satellite System [ABRA 72, BUTT 74], the use of a satellite channel will be assumed in all the numerical examples in this dissertation. In the next section, we shall examine some satellite channel characteristics and cost trends. Finally, in the last section, abstract models for the channel and channel users will be given.

2.1 Advantages of Satellite and Radio Packet Communications

Consider the use of packet communication in a computer-communication network environment to support large populations of (bursty) users over wide areas. We can identify the following advantages of satellite and ground radio channels over conventional wire communications:

(1) Elimination of complex topological design and routing problems

Topological design and routing problems are very complex in large networks [FRAN 72B, GERL 73]. Existing implementations suitable for a (say) 50 node network may become totally inappropriate for a 500 node network required to perform the same functions [FRAN 73]. On the other hand, ground radio and satellite channels used in the multi-access broadcast mode provide a completely connected network topology, since every user may access any other user covered by the broadcast.

(2) Wide geographical areas

Wire communications become expensive over long distances (e.g., transcontinental, transoceanic). Even on a local level, the communication cost for an interactive user on an alphanumeric console over distances of over 100 miles may easily exceed the cost of computation [ABRA 70]. On the other hand, satellite and radio communications are relatively distance-independent.

(3) Mobility of users

Since radio is a multi-access broadcast medium, it is possible for users to move around freely. This consideration will soon become important in the development of personal terminals in future telecommunication systems [MART 71, ROBE 72A].

(4) Large population of active and inactive users

In wire communications, the system overhead usually increases directly with the number of users (e.g., polling schemes). The maximum number of users is often bounded by some hardware limitation

(e.g., the fan-in of a communications processor). In radio communication, since each user is merely represented by an ID number, the number of active users is bounded only by the channel capacity and there is no limitation to the number of inactive (but potentially active) users.

(5) Flexibility in system design

A radio packet communication system can become operational with two or three users. The size of the user population can be increased up to the channel capacity. More users can be accommodated by increasing the radio channel bandwidth. In other words, the communication system can be made bigger or smaller without major changes in the basic system design and operational schemes.

(6) Statistical load averaging

In wire communications, the use of adaptive routing techniques [FULT 72] in a store-and-forward packet switched network, for example, enables communication links to be better utilized than in a circuit switched network. However, at any instant, there may still be unused channel capacity in some parts while congestion exists in other parts of the network. The application of packet switching techniques to a single high-speed satellite or radio channel permits the total demand of all user input sources to be statistically averaged at the channel. Note also that each user transmits data at the (high-speed) channel rate.

(7) Multi-access broadcast capability

This capability in radio communication may be useful for certain multi-point to multi-point communication applications.

(8) Reliability

The norminal bit error rate of a satellite channel using forward error correction techniques is estimated to be $P_{be} = 1 \times 10^{-9}$ and better, compared to $P_{be} = 1 \times 10^{-5}$ for typical terrestrial links [CACC 74].

2.2 Satellite Channel Characteristics and Cost Trends

In addition to their multi-access broadcast capability, satellite channels have other characteristics which distinguish them from conventional communication channels and must be taken into consideration in any satellite communication system design.

The satellite

We quote the following information on the Anik satellites [GRAY 74]:

> "The satellites are about 6 feet in diameter and 11 feet high. At launch they weighed about 1250 lbs. and their orbiting weight is about 600 lbs. Each satellite's electronics system is powered normally by about 23,000 solar cells with sufficient on-board battery capability to provide power during eclipse periods when the satellite is in shadow....The life expectancy of the batteries is a minimum of seven years. Each spacecraft consists of an electronic communications system, literally a microwave receiving and transmitting station in space, and on-board propulsion systems to inject it into its synchronous orbit and correct for wobble or spin."

Round-trip delay (RTD)

A satellite in geosynchronous orbit is stationed approximately 36,000 kilometers above the equator. A signal transmitted by an earth station to the satellite transponder (at one frequency) is beamed back to earth (at another frequency) and can be received by all stations covered by the transponder beam. The round-trip propagation delay (RTD) is approximately a quarter of a second. Depending on a station's geographical location on earth, a difference of 15 milliseconds exists. Furthermore, the satellite drifts approximately 200 miles in range during the day, which produces an additional two milliseconds difference in RTD. Without loss of generality, we shall assume the maximum RTD value for all stations in our work.

Burst synchronization and channel slotting

Despite differences in the RTD values of earth stations, tests performed with an Experimental TDMA system over INTELSAT I (Early Bird) during August 1966, indicate that transmission bursts from different stations can be synchronized at the satellite transponder requiring guard times less than 200 nanoseconds [GABB 68]. In our case of a packet switched system, the satellite transponder time was assumed to be the global reference time (channel time) for all earth stations. The very small guard time required for burst (packet transmission) synchronization demonstrates the feasibility of channel slotting. Several slotting techniques have been examined by Rettberg [RETT 73A].

Automatic acknowledgment

To ensure data integrity in a communication channel, a very reliable method is the use of an error detecting block code in conjunction with positive acknowledgment of each message by its recipient. In a satellite channel, any signal relayed by the transponder is received by all earth stations including the sender(s). Channel collision (packet transmissions overlapping in time at the satellite) will be known to the sender as well as the addressed receiver of a collided packet. Thus, assuming that the satellite channel has a low (random noise) error rate, positive acknowledgments may not be necessary.

Data rates and small earth stations

An excellent introduction to the currently operational SPADE system (using an INTELSAT IV global-beam transponder) is available in [CACC 71]. We summarize here some relevant information on channel data rates and considerations for small earth station operation.

The SPADE system utilizes single-channel-per-voice-carrier transmissions. 7-bit PCM encoding is used for voice with the encoded output at 56 KBPS (8000 samples/sec.). The channel transmitted bit rate is 64 KBPS. Since 4-phase coherent PSK modulation is used, the transmitted symbol rate is 32,000 symbols per second using a bandwidth of 38 KHz. The SPADE channel unit can be operated in continuous or voice-activated mode depending on whether data or voice is transmitted.

The SPADE system with standard INTELSAT earth stations will achieve a maximum capacity of 800 voice channels (assuming voice activation). This capacity is simultaneously bandwidth and power limited. Hence, if smaller earth stations (i.e., stations with smaller antennas) are used, the capacity will be power limited and there will be a reduction in system capacity. One approach to minimize the power limited condition is to use error coding to provide a tradeoff of the excess available bandwidth to reduce the net per-channel required power.

Costs and other considerations

We emphasize again that we are primarily interested in systems involving fairly large populations of users. In such a packet switched satellite broadcast system, the cost of earth stations dominates the satellite bandwidth cost. A standard INTELSAT earth station with a 97-foot antenna costs between $3-3.5 million dollars! We note that if a node has enough traffic to justify the cost of a large satellite station, its traffic is probably high enough and consequently, sufficiently "smooth" to warrant its own satellite channel. On the other hand, an earth station for a domestic satellite system (such as Anik and future U. S. systems) can use a 30-foot antenna which costs from $150,000 upwards.* This figure is comparable to the costs of peripheral devices in present large computer installations. In a recent study [DUNN 74], even smaller earth stations

*The above figures were quoted in an informal conversation with people in the General Electric Company Space Division.

(with antenna diameter between 10 to 15 feet) were suggested. The annual cost per station was estimated to be approximately $5,000 to $15,000.

We also note that there is an existing regulatory restriction on the use of an INTELSAT IV channel in the multi-access broadcast mode for several stations. Discussions are under way with various agencies to remove these regulatory barriers in either the INTELSAT system or one of the domestic systems [ABRA 73].

With domestic satellite systems, data rates are not limited to that of a single voice channel. For example, data rates ranging up to 60 MBPS will be available over the American Satellite Corporation system. Furthermore, specialized network configurations will be available to suit a user's customized requirements [CACC 74].

We quote the following remarks on projected satellite technology cost trends by Roberts [ROBE 74]:

> "Although terrestrial communications cost appears to limit the future price of computer-communication service, including packet-switching networks, the situation is rapidly changing with the introduction of domestic satellites....Applying the least-mean-square exponential fit to this data, the rate of technological improvement in the cost performance of satellites is found to be 40.7 percent per year, or a factor of ten every 6.7 years. This can only be treated as a crude estimate of the cost trend for satellite communication, but since it is quite in keeping with the general cost trend for electronics, it is a

quite credible growth rate....Satellites will play an important role in reducing the future cost of packet switching service...."

2.3 An Abstract Model

Consider packet switched satellite and radio systems using the slotted ALOHA random access technique. In order to evaluate the performance of these communication systems via model building and theoretical analysis or simulation, it is desirable to define abstract models which include only the salient properties and operational features. We define the following models for the multi-access broadcast channel and its users.

2.3.1 The Channel

We assume a bandwidth limited channel. Since users of this channel are in general geographically distributed, we assume a global reference time called channel time (see Section 1.3). Channel transmissions are assumed to be free of random noise errors so that a packet of data is received incorrectly if and only if it collided with another packet at the channel. We assume fixed size packets. Channel time is slotted such that all users synchronize their packet transmissions into channel slots. A channel slot length is exactly equal to the duration of a packet transmission. Any guard time required to separate packet transmissions in the channel is neglected. From now on, time will be expressed in channel slots. All rates will be normalized with respect to a channel time slot.

Channel Input

The channel input in a channel time slot is a random variable representing the total number of new packets transmitted by all users in that time slot. The channel input rate S is the average number of new packet transmissions per time slot (assuming stationary conditions).

Channel Traffic

The channel traffic in a channel time slot is a random variable representing the total number of packet transmissions (both new and previously collided packets) by all users in that time slot. The channel traffic rate G is the average number of packet transmissions per time slot (assuming stationary conditions).

Channel Throughput (Output)

The channel throughput (or output) in a channel slot is a random variable representing the number (0 or 1) of successful packet transmissions in that time slot. The channel throughput (output) rate S_{out} is the same as the probability of the channel traffic in a channel slot exactly equal to one (assuming stationary conditions). The maximum possible throughput rate of a channel is defined to be the channel capacity S_{max}.

Retransmission Delay (RD)

Whenever a packet has an unsuccessful transmission, it incurs a retransmission delay equal to the amount of time from the packet's collision at the channel until its subsequent retransmission attempt. Each retransmission delay can be regarded as the sum of a deterministic

delay and a random delay. Random delays are needed since if packets which collided at the channel are retransmitted after the same deterministic delay, they will collide again for sure. (Of course, if there is only a small number of channel users, each user may use a separate deterministic RD and no random delay is necessary.)

For example, in a satellite system, the deterministic delay corresponds to a station's round-trip propagation delay (assumed to be the same for all earth stations). Random delays may be inserted independently by earth stations into the retransmission times of previously collided packets to minimize their probability of colliding again. In a terminal access radio communication network such as the ALOHA system, the retransmission delay corresponds to a terminal's positive acknowledgement time-out interval.

The retransmission delay is probably the most important design variable in the system. As we shall see, it determines the channel's throughput-delay performance, dynamic and stability behavior. As a result, it will be utilized for dynamic channel control. We shall assume the deterministic delay in RD to be R slots* and the random delay to be uniformly distributed over K slots. This will be referred to as uniform retransmission randomization.** Hence, RD has the probability density function shown in Fig. 2-1.

---

*In a terminal access ground radio system, the round-trip propagation delay(corresponding to the deterministic delay) is in general a fraction of a time slot rather than equal to many slots.

**Another simple probability density function which can be utilized is the geometric distribution (geometric retransmission randomization). It turns out, as we show in Chapter 5 via simulations, that the channel performance is dependent primarily upon the average value of RD and quite insensitive to its exact distribution.

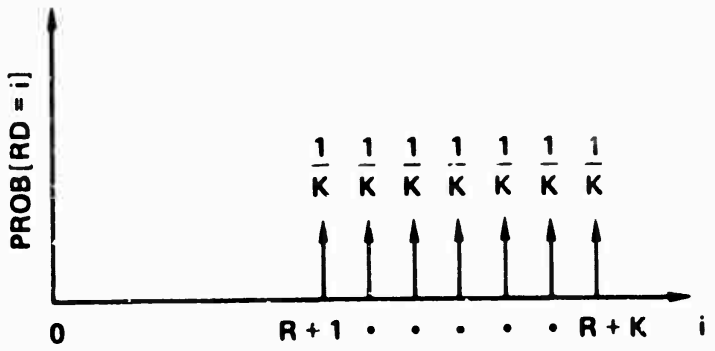


Figure 2-1. Probability Density Function for a Retransmission Delay (RD).

Packet Delay

The total delay a packet incurs is defined to be the amount of time from the packet's initial transmission until "successful transmission occurs." (Nodal processing delays will be neglected.) Conditioning on a successful packet transmission, let R' be the delay from the time the sender of the packet finishes transmitting the packet until successful transmission occurs. In a satellite channel, this amount of time is just one round-trip propagation delay (hence, $R' = R$). In a ground radio terminal access network, the meaning of R' is not so well defined; it can either be interpreted as the channel propagation time from the terminal to the central computer or as the delay until a positive acknowledgment is received from the central computer. Without loss of generality, we shall assume $R' = R$ throughout this dissertation. We show in Fig. 2-2 the total delay of a packet which has exactly one collision.

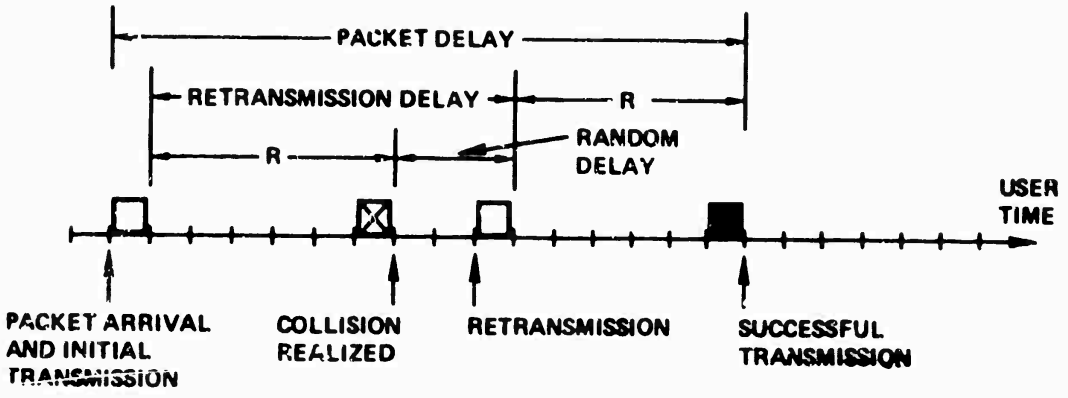


Figure 2-2. Delays Incurred by a Small User Packet

Numerical Constants

For purposes of numerical examples throughout this dissertation, we assume the following numerical constants based upon a satellite voice channel. A satellite channel is characterized by a very large channel propagation delay (compared to ground radio). These assumptions will be reflected in our numerical results and conclusions drawn from these results. However, the methodology and analytic tools developed in this dissertation will not be dependent upon these assumptions.

Unless stated otherwise, R will be taken to be 12 channel time slots and each time slot is 22.5 milliseconds long, giving 44.4 slots/second. The above figures are computed from the assumptions of a 50 KBPS satellite voice channel,* 1125 bits/packet (including

*Actually, a SPADE voice channel has a transmitted bit rate of 64 KBPS in which case the assumption of 1440 bits/packet (probably including error correcting codes, guard time, etc.) will give rise to the same numerical constants.

overhead bits for address, parity check, etc.) and a round-trip propagation delay of 0.27 second.

Note that we have assumed a 50 KBPS channel because this happens to be a currently available satellite channel data rate. With the introduction of domestic satellite systems which will provide a wider range of data rates [CACC 74], a higher channel data rate may be considered (e.g., a 1.5 MBPS channel was included in the proposed Telenet packet switching network [TELE 73]). On the other hand, we may want to use a lower data rate for a ground radio system.

2.3.2 Channel Users

"Users" are defined to be entities which have the capability (e.g., antenna, transceiver, modem, logic, buffers, etc.) to transmit and receive packets of information over the channel as well as to accept input and deliver output to its "source." Examples of channel users may include a wide variety of devices such as hand-held personal terminals [ROBE 72A], teletype consoles, data concentrators and nodal switching units (see Fig. 1-1). The terminal control units of the ALOHA system [KUO 73] and the satellite IMPs of the ARPA network [BUTT 74] are some practical examples.

In this dissertation, we shall distinguish two abstract models of users: small users and large users.

Small users

A small user is one with buffer space for exactly one packet awaiting transmission. If and only if the buffer is empty, a packet arrival occurs with probability σ. (A packet arrival is said to

occur only when a new packet is ready for transmission in the current time slot, i.e., after it has been entered by the source and processed by the user.) Thus, the user "think" time (i.e., the time between the successful transmission of a packet and the initial transmission of the next packet) is geometrically distributed with an average value of $\frac{1}{\sigma}$ slots. A small user can be in one of two states: blocked (buffer occupied) or thinking (buffer empty). An example of a small user in a ground radio system is a teletype console with keyboard lockout such that the human user cannot enter a new line of characters (a packet) before the previous packet is successfully transmitted.

A small user or terminal as characterized by our abstract model may or may not be "small" in a real system. If, instead of a 50 KBPS channel, we now consider a 2 MBPS channel with 20 kilobit packets and if the sum of the average user think time and packet delay is 2 seconds, the "small" user has a data rate of 10 KBPS!

Large users

Large users will be considered in Chapter 3 only. A large user is defined to be one with a large buffer capacity such that new packets generated by the source will never be blocked due to lack of buffer space. Unless stated otherwise, the stream of packet arrivals to a large user is assumed to be a Poisson process.

In a large user, several packets may be awaiting transmission at the same time. We assume that all new arrivals are scheduled for transmission immediately. A scheduling conflict occurs when more than

one packet is scheduled to transmit in the current slot. The highest priority packet will transmit while the other packets are rescheduled independently (see below). Any priority rule will give rise to the same average packet delay (conservation law! [KLEI 64]). The following priority rule will be assumed for mathematical convenience.

Priority rule

We list in decreasing order of priority (depending on a packet's most recent history) for transmitting in the current slot:

(1) packets randomized into the current slot after a collision at the channel

(2) packets randomized into the current slot after a scheduling conflict

(3) new arrivals in the current slot

The first-come-first-served rule is used for packets in the same priority group. Ties are broken by random selection.

Rescheduling delay

A packet which is blocked due to a scheduling conflict is rescheduled in one of the next L slots, each such slot being chosen with probability $\frac{1}{L}$ (uniform rescheduling randomization). The average rescheduling delay is thus $(L + 1)/2$. We note that the uniform rescheduling randomization serves the same purpose as the uniform retransmission randomization. Our numerical results in this dissertation will be obtained using the same value for both L and K. We show below in Fig. 2-3 the total delay of a large user packet which has one channel collision and is rescheduled three times.

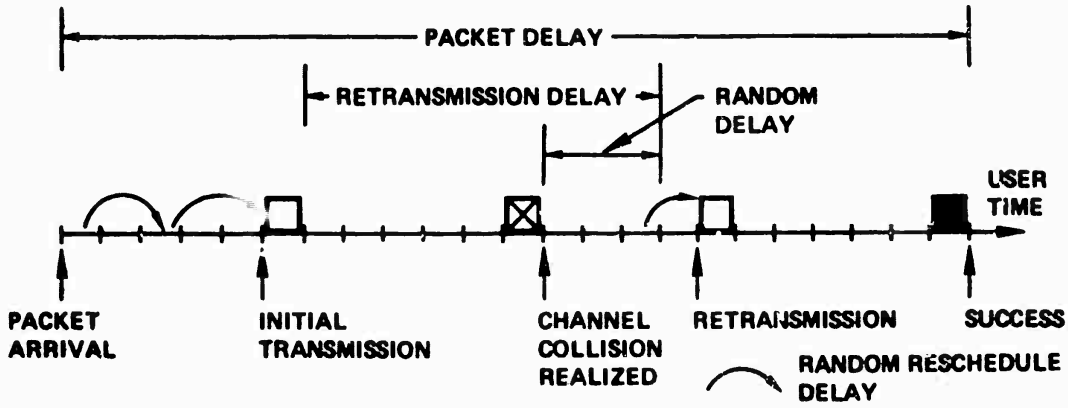


Figure 2-3. Delays Incurred by a Large User Packet.

CHAPTER 3

THROUGHPUT-DELAY PERFORMANCE

3.1 Introduction

In this chapter, analytic models are developed to predict the throughput-delay performance of the slotted ALOHA channel described in the last chapter. A gamut of throughput-delay tradeoffs will be presented corresponding to

- the infinite population model in which the channel supports input from a large number of small users modeled as a Poisson channel input source
- the large user model in which the channel user population consists of a large user (with buffering and scheduling capabilities) in addition to the population of small users above
- the finite population model in which the channel user population consists of a small number of large users

Small and large users may correspond to any physical devices which satisfy their abstract model descriptions given in Section 2.3.2. For example, a small user may represent a teletype console in a ground radio environment or an earth station in satellite communications as long as such a "small" user generates (independently) a new packet for transmission over the multi-access broadcast channel only after its previous packet has been successfully transmitted.

We show below that the slotted ALOHA channel capacity for the infinite population model is less than 37 percent. However, when a major fraction of the channel input is from a single large user which can buff

and schedule its own conflicting demands, both the channel capacity and throughput-delay performance can be significantly improved. Such improvements are also possible with a channel user population consisting of a small number of large users. However, when the number of large users is ten or more, we show that the channel throughput-delay results already approximate those of an infinite population model.

Throughput-delay results in this chapter are obtained under the assumption of equilibrium conditions. Monte Carlo simulations indicate that often this assumption is valid only for some finite time period beyond which the channel goes into "saturation." This phenomenon will be characterized in Chapter 5. The possibility of unstable channel behavior was first brought up in a private conversation with Martin Graham (University of California, Berkeley).

3.2 The Infinite Population Model

3.2.1 Assumptions

An abstract model for the slotted ALOHA channel is given in Section 2.3.1. We assume here that the user population consists of a very large number of small users such that V^t, the channel input in the t^{th} slot, is an independent process and has a stationary Poisson distribution with an average of S packets/slot.

Suppose X^t is the channel traffic in the t^{th} time slot. We shall assume that during the time period of interest X^t

(1) is an independent process,

(2) is Poisson distributed, and

(3) has stationary probability distribution.

These assumptions will be referred to as the independence assumption, the Poisson assumption, and the stationarity assumption, respectively.

We define equilibrium solutions (equilibrium points, equilibrium contour) to be those values of the channel input rate S and the channel traffic rate G such that the condition, channel throughput rate equal to the channel input rate, is satisifed. In this chapter, we shall be concerned only with equilibrium solutions. The channel is said to be in equilibrium at an equilibrium point during a period of time if the channel traffic X^t is a stationary process and the average channel traffic and throughput given by the stationary distribution of X^t satisfy the equilibrium point.

We show in Chapter 5, that slotted ALOHA channels supporting input from a large but finite number of small users are either "stable" or "unstable." For stable channels, the equilibrium throughput-delay tradeoffs given in this chapter are achievable over an infinite time horizon. On the other hand, an unstable channel will go into "saturation" after some finite time period.

Both the independence and Poisson assumptions represent approximations in our analytic model. Their accuracy will be examined by comparing analytic results with results from Monte Carlo simulations. Further tests to examine the Poisson assumption are given in Appendix A. It will also be shown in Chapter 4 that the Poisson assumption is actually implied by the independence assumption when the uniform randomization interval K is large.

3.2.2 The Analysis

Let E be the average number of retransmissions a packet incurs. Consider the time interval $[t_0, t_1]$ during which

$$\sum_{t=t_0}^{t_1} X^t = \text{total number of packet transmissions in } [t_0,t_1]$$

and

$$\sum_{t=t_0}^{t_1} \Delta(X^t) = \text{total number of successful packet transmissions in } [t_0,t_1]$$

where

$$\Delta(y) = \begin{cases} 1 & y \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

Under the independence and stationarity assumptions, the average number of transmissions required for a packet is

$$1 + E = \lim_{(t_1 - t_0) \to \infty} \frac{\sum_{t=t_0}^{t_1} X^t/(t_1 - t_0 + 1)}{\sum_{t=t_0}^{t_1} \Delta(X^t)/(t_1 - t_0 + 1)} = \frac{G}{S_{out}}$$

For an equilibrium solution, the channel throughput rate S_{out} is equal to the channel input rate. Thus,

$$1 + E = \frac{G}{S} \tag{3.1}$$

We next define q to be the probability of success given that a packet transmission has occurred. By similar arguments to the above, we have

$$q = \frac{S}{G} = \frac{1}{1 + E} \tag{3.2}$$

The slotted ALOHA channel capacity for the infinite population model can be obtained by the following zeroth order approximation approach similar to Abramson's analysis of an unslotted ALOHA channel [ABRA 70, ROBE 72B]. Consider a test packet transmission in a channel time slot. Its probability of success is the probability that no other packet is transmitted in the same channel slot. Applying the Poisson assumption and Eq. (3.2), we have

$$q = e^{-G} \tag{3.3}$$

and

$$S = Ge^{-G} \tag{3.4}$$

Now if we differentiate Eq. (3.4) with respect to G, it can easily be shown that the maximum channel throughput rate (channel capacity) is

$$S_{max} = \frac{1}{e} \cong 0.368$$

The zeroth order approximation above disregards both the time history of the test packet and the uniform randomization interval K for retransmissions. In order to compute the average packet delay D, we shall take the following approach (to be referred to as the first order approximation).

Given a test packet, two states are distinguished depending upon its immediate history: new or previously collided. We then define

q_n = Prob[success/transmission of a new test packet]

q_t = Prob[success/transmission of a previously collided test packet]

Hence,

Prob[a packet is retransmitted exactly i times before success]

$$= (1 - q_n)(1 - q_t)^{i-1} q_t \qquad i \geq 1$$

$$E = \sum_{i=1}^{\infty} i(1 - q_n)(1 - q_t)^{i-1} q_t$$

$$= \frac{1 - q_n}{q_t} \tag{3.5}$$

and

$$q = \frac{1}{1 + E} = \frac{q_t}{q_t + 1 - q_n} \tag{3.6}$$

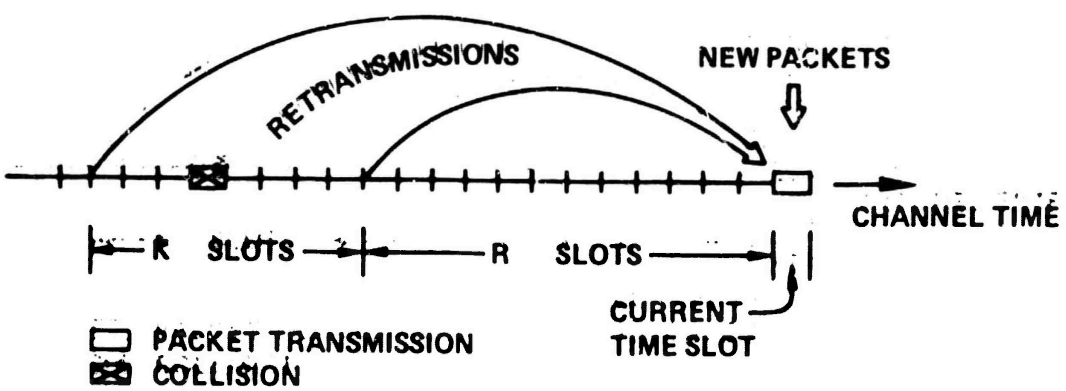


Figure 3-1. Channel Traffic Into a Time Slot.

We now condition on a test packet transmission in the current time slot. This transmission may be unsuccessful due to interference by new or previously collided packets transmitting also in the current slot (see Fig. 3-1). Suppose the test packet had a previous collision in one of the K slots (say the j^{th}) indicated in the figure. We define q_c to be the conditional probability that no packet from the j^{th} slot other than the test packet retransmits into the current slot. Using the Poisson assumption for channel traffic in each of the K slots,

$$q_c = \frac{1}{1 - e^{-G}} \left[\sum_{n=1}^{\infty} \left(\frac{K - 1}{K} \right)^n \frac{G^n}{n!} e^{-G} \right]$$

$$= \frac{e^{-\frac{G}{K}} - e^{-G}}{1 - e^{-G}}$$

Let q_0 be the probability that no packet which collided in one of the $(K - 1)$ slots (other than the j^{th} slot) retransmits in the current slot. We have

$$q_0 = \sum_{n=2}^{\infty} \left(\frac{K - 1}{K} \right)^n \frac{G^n}{n!} e^{-G} + Ge^{-G} + e^{-G}$$

$$= e^{-\frac{G}{K}} + \frac{G}{K} e^{-G}$$

Invoking the independence assumption, we then obtain

$$q_t = q_c \, q_0^{K-1} \, e^{-S}$$

$$= \frac{e^{-\frac{G}{K}} - e^{-G}}{1 - e^{-G}} \left[e^{-\frac{G}{K}} + \frac{G}{K} e^{-G} \right]^{K-1} e^{-S} \qquad (3.7)$$

Now suppose the test packet is a new packet. By similar arguments to the above, we can express its probability of success as

$$q_n = q_0^K \, e^{-S}$$

$$= \left[e^{-\frac{G}{K}} + \frac{G}{K} e^{-G} \right]^K e^{-S} \qquad (3.8)$$

From Eqs. (3.2) and (3.6), we then have

$$S = G \frac{q_t}{q_t + 1 - q_n} \qquad (3.9)$$

The <u>average delay D</u> incurred by a packet at the channel includes the channel propagation time, the packet transmission time and retransmission delays and is given below (in number of time slots) by

$$D = R + 1 + E\left(R + \frac{K+1}{2}\right) \qquad (3.10)$$

where $R + (K + 1)/2$ is equal to the average retransmission delay (see Fig. 2-1).

Equations (3.7)-(3.9) form a set of nonlinear implicit equations which must be solved numerically for the equilibrium relationships between S and G. The average packet delay can then be obtained from Eqs. (3.5) and (3.10). Numerical solutions will be given in the next section. Below we examine some limiting cases in which explicit solutions are available and consider their implications.

Limiting results as $K \to \infty$

It can easily be shown from Eqs. (3.7)-(3.9) that in the limit as $K \to \infty$,

$$\lim_{K\to\infty} \frac{S}{G} = \lim_{K\to\infty} q_n = \lim_{K\to\infty} q_t = e^{-G} \tag{3.11}$$

These limiting results are consistent with the Poisson assumption we made. In fact, in the next chapter we show, given only the independence assumption, that in the limit as $K \to \infty$, the channel traffic in a time slot must be Poisson distributed.

Observe that Eqs. (3.11) are the same as the zeroth order approximation results. Thus, the first order approximation reduces to the zeroth order approximation in the limit as $K \to \infty$ (which corresponds to infinite average packet delay!).

Limiting results as $S \to 0$

In the limit as the channel input rate S decreases to zero, Eqs. (3.7)-(3.10) reduce to

$$\lim_{S\downarrow 0} \frac{S}{G} = \lim_{S\downarrow 0} q_n = 1 \tag{3.12}$$

$$\lim_{S\downarrow 0} q_t = \frac{K-1}{K} \tag{3.13}$$

and

$$\lim_{S \downarrow 0} D = R + 1 \tag{3.14}$$

Define K_{opt} to be the value of K minimizing the average packet delay D for a fixed channel input (throughput) rate S.

Proposition 3.1 In the limit as $S \downarrow 0$, D is convex in K and K_{opt} is given by the largest integer K such that

$$K^2 - 3K - 2R \leq 0 \tag{3.15}$$

Proof With $K = 1$, any channel collision will propagate indefinitely. Thus, $K = 1$ cannot be optimal. We shall consider $K \geq 2$. For an arbitrarily small S, Eqs. (3.7) and (3.8) become

$$q_t \cong \frac{K - 1}{K} (1 - S)$$

$$q_n \cong (1 - S)$$

and from Eqs. (3.5) and (3.10),

$$E = \frac{1 - q_n}{q_t} \cong \frac{S}{\frac{K - 1}{K} (1 - S)} \cong \frac{S}{\frac{K - 1}{K}}$$

$$D \cong R + 1 + \frac{S}{\frac{K - 1}{K}} \left[R + 1 + \frac{K - 1}{2} \right]$$

Since $S > 0$, D is minimized by minimizing the function

$$f(K) = \frac{K}{K - 1} (R + 1) + \frac{K}{2}$$

Consider

$$f(K) - f(K - 1) = - \frac{(R + 1)}{(K - 1)(K - 2)} + \frac{1}{2} \qquad K > 2$$

which is less than or equal to zero if $K^2 - 3K - 2R \leq 0$. Now consider

$$[f(K + 2) - f(K + 1)] - [f(K + 1) - f(K)]$$

$$= \frac{2(R + 1)}{(K + 1) K(K - 1)} > 0 \qquad K \geq 2$$

which implies that $f(K)$ is convex in K .

From the above results, D is convex in K and minimized by the largest integer K such that $K^2 - 3K - 2R \leq 0$.

Q.E.D.

For $R = 12$, $\lim_{S \to 0} K_{opt} = 6$ which, as we show below, represents a lower bound on the optimum value of K for any channel input rate S .

3.2.3 Throughput-Delay Results

Numerical results

Equations (3.7)-(3.9) were solved numerically and the results plotted in Figs. 3-2 and 3-3. In Fig. 3-2, whe show the probability of success, q , as a function of K at a different channel traffic rates. For a fixed G , q increases with K and rapidly approaches its limiting value of e^{-G} as predicted by Eq. (3.11). q also increases as G decreases for a fixed K .

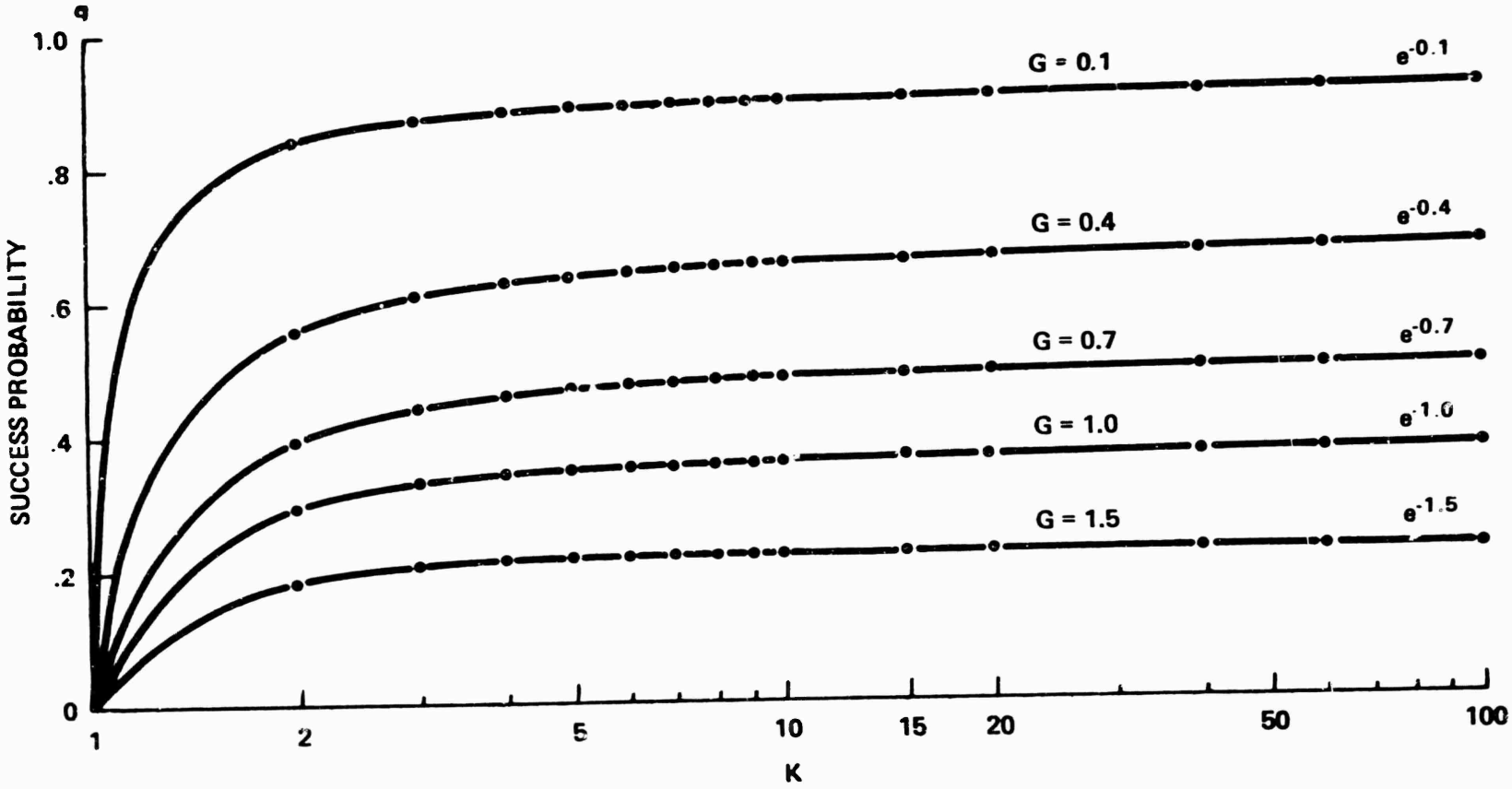


Figure 3-2. Probability of Success as a Function of K.

In Fig. 3-3, the channel throughput rate (same as the channel input rate S in an equilibrium solution) is shown as a function of G for fixed values of K. For a fixed G, the channel throughput rate increases rapidly to its limiting value of Ge^{-G} as K increases to infinity. (Note that for $K = 15$ it is almost there.) The maximum channel throughput rate occurs at $G = 1$ for each K and the channel capacity $S_{max} = e^{-1}$ in the $K \to \infty$ limit.

The average packet delay D is computed using Eq. (3.10) (and assuming $R = 12$). In Fig. 3-3, we plotted the loci of several constant delay values in the S,G plane. Note the way these loci bend over sharply defining a maximum channel throughput rate for a fixed value of D; observe the cost in channel throughput if we want to limit the average packet delay. This effect is clearly seen in Fig. 3-4, which is the fundamental display of the <u>throughput-delay tradeoff</u> for the infinite population model. This figure shows the throughput-delay equilibrium contours for fixed values of K. The minimum envelope of these contours defines a <u>tight lower bound</u> on throughput-delay performance for this system and thus, represents the <u>optimum channel performance</u> for the infinite population model. Considering this optimum curve, we note how sharply the average packet delay increases near the maximum channel throughput rate $S_{max} = 0.368$; it is clear that an extreme price in delay must be paid here for an infinitesimal incremental gain in throughput. Also shown in this figure are the constant G contours. Thus, Figs. 3-3 and 3-4 are two alternate displays of the relationship among the four critical system variables S, G, D and K under equilibrium conditions.

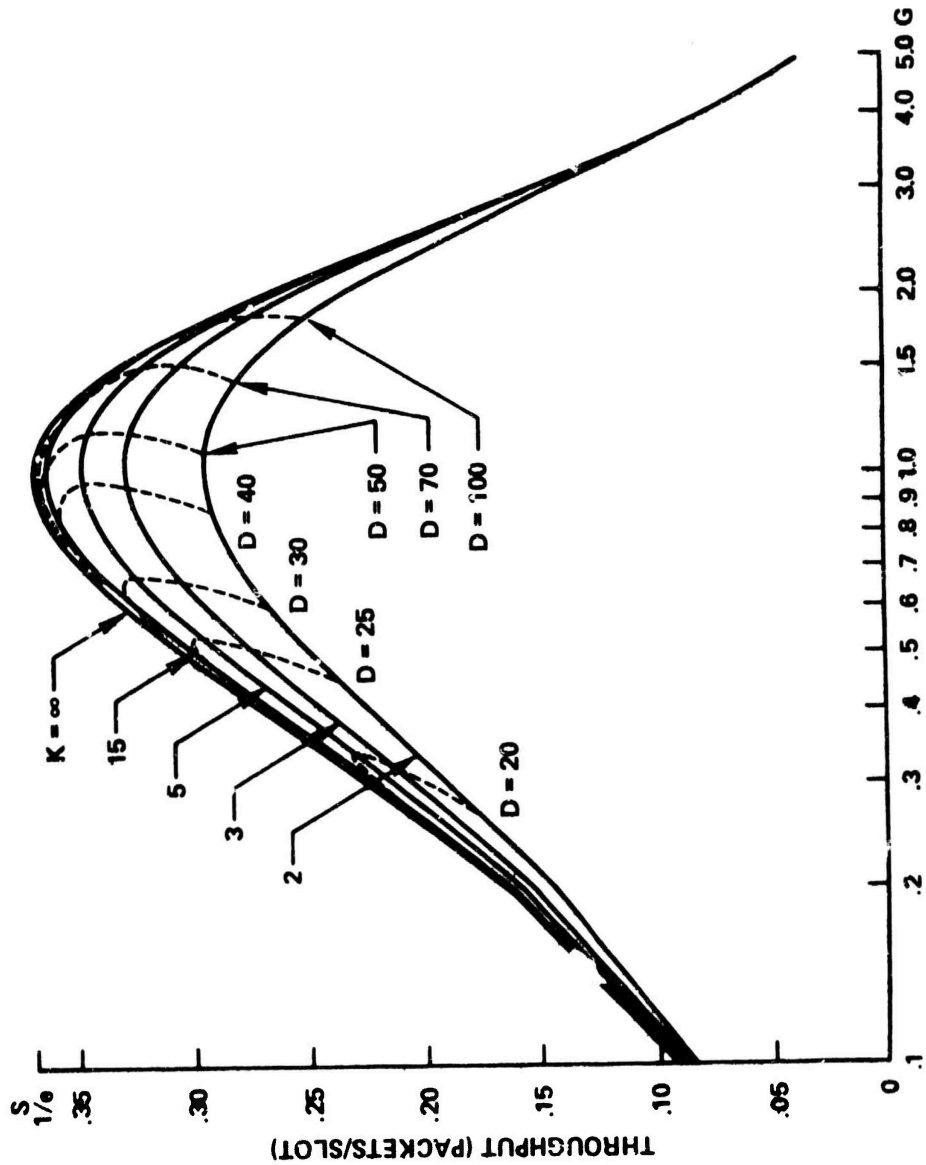


Figure 3-3. S Versus G

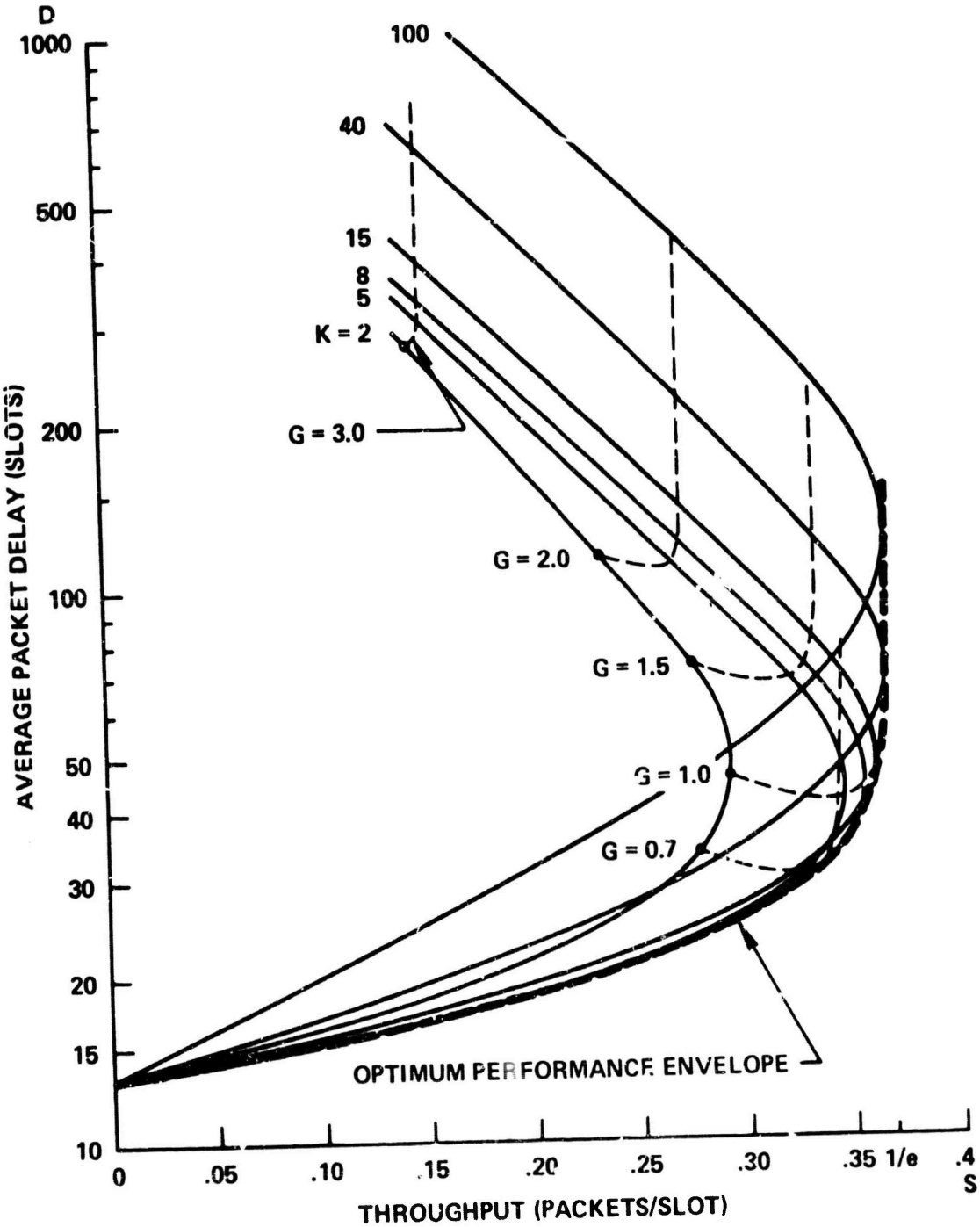


Figure 3-4. Throughput-Delay Tradeoff.

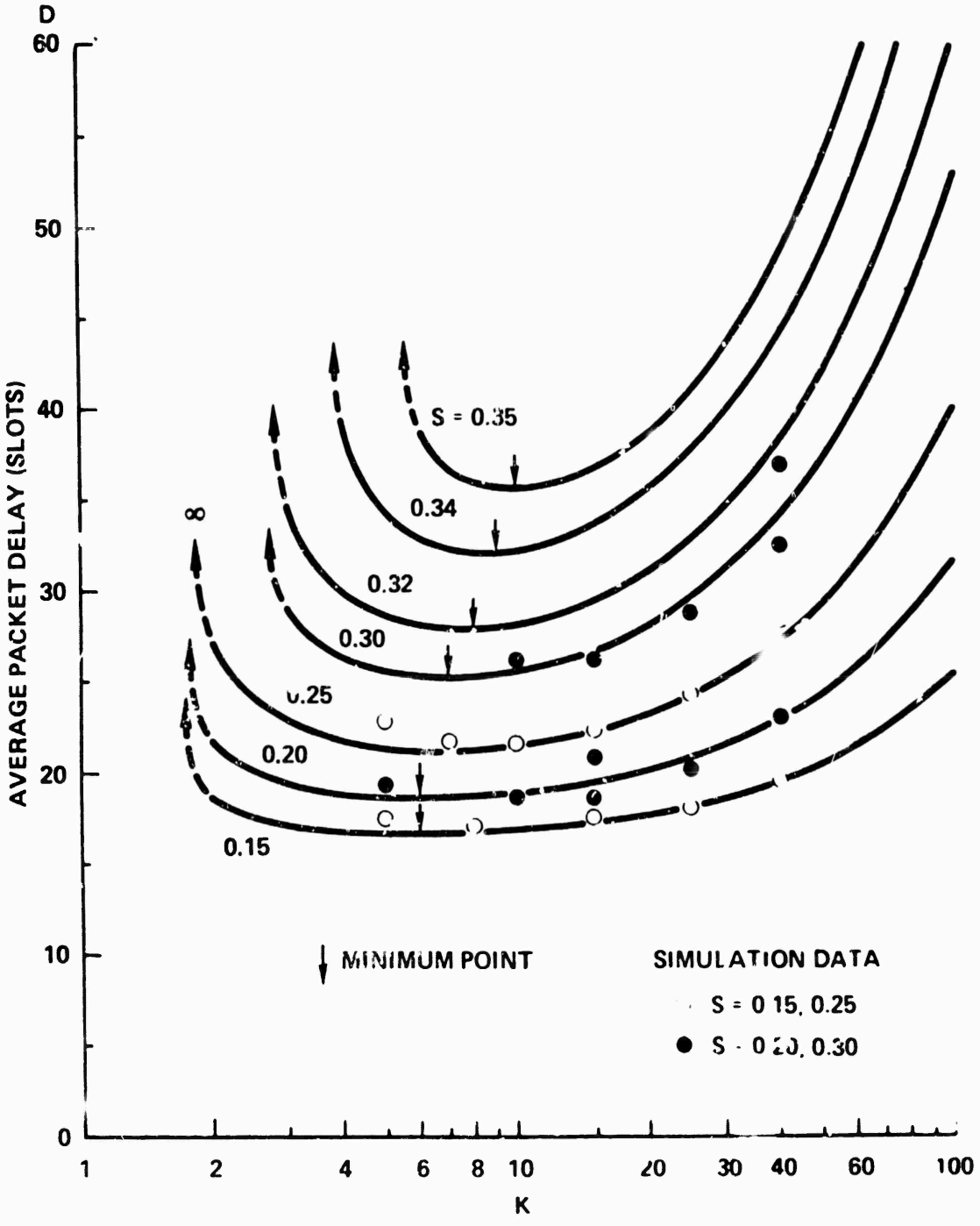


Figure 3-5 Average Packet Delay Versus K

In Fig. 3-5, the average packet delay is plotted as a function of K for constant values of S. For a fixed S, the curve is quite flat near K_{opt}. Thus, a K value much bigger than K_{opt} can be used without increasing D appreciably. A large K is preferable since it increases the maximum channel throughput rate and improves channel stability (as we shall see in Chapter 5). In Fig. 3-6, we show K_{opt} as a function of S. Note that K_{opt} is a nondecreasing function in S and is bounded below by 6 as $S \to 0$, which is predicted by Eq. (3.15) for $R = 12$.

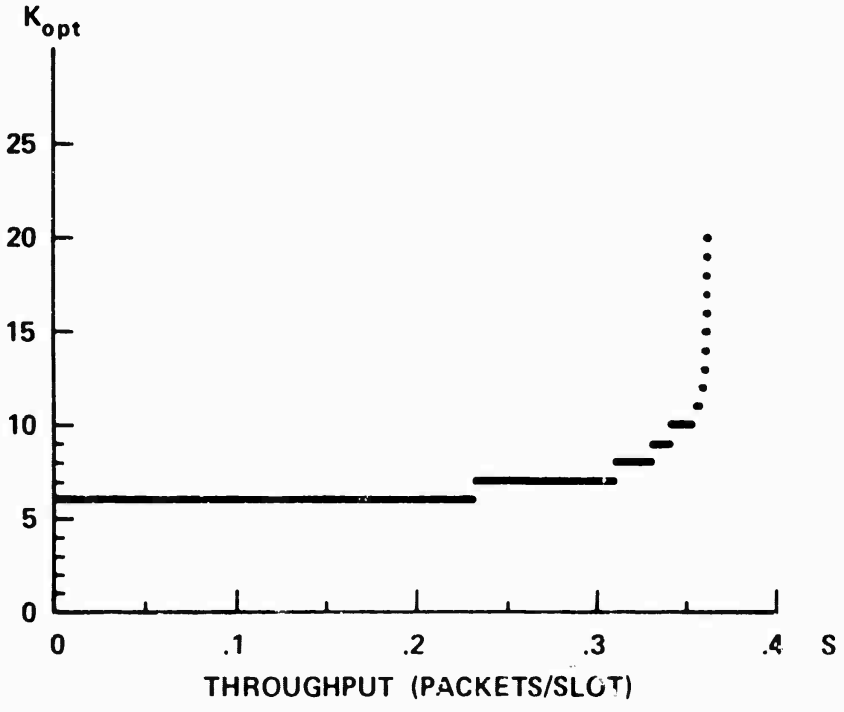


Figure 3-6. K_{opt} Versus S.

Simulations

A simulation program was developed to test the accuracy of the approximations introduced by assumptions in the above analysis. In the simulation program, new packets are generated from a Poisson distribution at a constant rate S which, together with the uniform randomization interval K, constitute the simulation input parameters. Packet delays are obtained by time-stamping each packet at the time of its creation. The exact delay a packet incurs can then be computed when it is successfully transmitted. Both long-term statistics for the duration of the simulation run and short-term statistics for consecutive time intervals (of, say, 400 slots each) are available. Short-term statistics serve to protray approximately the dynamic channel behavior.

Recall that the analytic results we have obtained so far are all based upon the assumption that the channel is in equilibrium. Referring to Fig. 3-4, we see that given S and K, there are two possible equilibrium solutions for D corresponding to a small delay value (say D_A) and a much larger delay value (say D_B). We shall refer to the equilibrium point given by S, K and D_A as the channel operating point, since this is the desired channel performance given S and K.

Each simulation run was observed to behave in the following manner. Starting from an initially empty system, the channel stays in equilibrium at the channel operating point for a finite period of time until stochastic fluctuations give rise to some high traffic rate which reduces the channel throughput rate which in turn further increases the channel traffic rate. As this vicious cycle continues,

the channel becomes "flooded" with collisions and retransmissions. The channel throughput rate vanishes rapidly to zero. This phenomenon will be referred to as channel saturation. (For the equilibrium point corresponding to D_B, no channel equilibrium for any length of time has been observed.) Thus, simulations indicate that we can assume channel equilibrium at the channel operating point, but only for some finite time period. Such time period is a random quantity and will be characterized in Chapter 5 as a measure of channel stability. The expected value of this random time period increases as K increases or S decreases. For a sufficiently small value of S or large value of K, the assumption of channel equilibrium was always valid for the simulation duration we considered. In Fig. 3-7, we show a simulation run for $S = 0.35$ and $K = 15$, which give rise to a relatively short duration of channel equilibrium. As we see in the figure, after 3000 time slots, the channel traffic rate increases very rapidly as the channel throughput rate decreases to zero.

In Fig. 3-5, simulation points are indicated. We show only those simulation runs in which the channel stays in equilibrium for the duration of the run (assumed to be 8000 slots). The (heuristic) criterion we used for channel equilibrium is that the average channel traffic in each of the short-term statistics intervals (400 slots each) must be less than one. Observe that the largest channel input rate used for these simulations is 0.3. For a larger input rate, our criterion of channel equilibrium is often not satisfied unless a very large K (say, $K = 60$) is used, which gives rise to a

SLOTTED SATELLITE COMMUNICATION SIMULATION USING POISSON DISTRIBUTED ARRIVAL GENERATION

INPUT PARAMETERS => => =>
INPUT RATE = 0.350
PROPAGATION DELAY = 12
K = 15

AVERAGE VALUES IN 200 TIME SLOT PERIODS

TIME PERIOD	THROUGHPUT RATE- S	TRAFFIC RATE- G	PACKET DELAY- D	FRACTION EMPTY	AVERAGE BACKLOG
1 - 200	0.330	0.510	17.924	0.595	3.1
201 - 400	0.375	0.605	30.892	0.530	5.3
401 - 600	0.360	0.860	32.764	0.425	9.5
601 - 800	0.340	0.840	43.147	0.435	10.5
801 - 1000	0.315	1.415	58.889	0.250	21.5
1001 - 1200	0.380	1.260	72.066	0.260	18.0
1201 - 1400	0.325	0.455	37.215	0.610	3.0
1401 - 1600	0.355	0.480	20.803	0.590	2.5
1601 - 1800	0.275	0.405	20.600	0.665	2.6
1801 - 2000	0.360	0.560	27.528	0.550	4.5
2001 - 2200	0.330	0.430	18.561	0.620	1.8
2201 - 2400	0.310	0.545	22.065	0.580	4.3
2401 - 2600	0.335	0.840	44.866	0.455	9.9
2601 - 2800	0.320	0.705	34.703	0.500	7.5
2801 - 3000	0.325	1.085	43.815	0.350	14.1
3001 - 3200	0.310	1.715	67.161	0.180	26.9
3201 - 3400	0.105	3.255	147.143	0.050	61.6
3401 - 3600	0.015	5.910	220.333	0.000	113.4
3601 - 3800	0.000	9.155	0.000	0.000	179.3
3801 - 4000	0.000	12.400	0.000	0.000	243.3

LONG TERM STATISTICS =>=>=>=>=>=>=>

THROUGHPUT = 0.270
TRAFFIC = 2.259
IDLE SLOTS = 0.371
AVERAGE DELAY = 41.287
AVERAGE BACKLOG = 38.9

TIME FOR LONG TERM STATS = 3600

Fig. 3-7 Simulation run with a short duration of channel equilibrium.

large average delay. Note that the simulation and analytic results agree very well, thus lending validity to approximations in our analysis (the independence assumption and the Poisson assumption for the channel traffic X^t). Further simulation results on the Poisson approximation are examined in Appendix A.

3.3 The Large User Model

3.3.1 The Large User Effect

The 1/e limitation on the capacity of a slotted ALOHA channel supporting input from a large number of small users (i.e., the infinite population model above) is due to the loss of all packets whenever simultaneous transmissions are made by two or more users. On the other hand, when the channel is dedicated to a single large user with buffering and scheduling capabilities, simultaneous demands from the large user's input sources can be queued up and served according to some priority rule.* In this case, a channel throughput rate arbitrarily close to unity can be achieved at the expense of a very large average packet delay. In fact, the absolute optimum throughput-delay tradeoff performance of the communication channel can be obtained by modeling it as a single server queue. Intermediate throughput-delay tradeoff performances are possible which lie between the two extremes of the infinite population model and the single server queueing model. A continuum of such intermediate tradeoff performances will be given below for the large user model in which the random access channel is shared by a large user and the small users of an infinite user

*We are only interested in the average packet delay which is independent of the exact priority rule as a result of the conservation law [KLEI 64].

population. Further intermediate tradeoff performances will be given in Section 3.4 below for a finite number of large users. In Fig. 3-8, we show a picture of the large user model in a possible satellite communications system. The large user model also represents a terminal access network in which a single radio channel is used for both terminal-to-computer and computer-to-terminal communications.

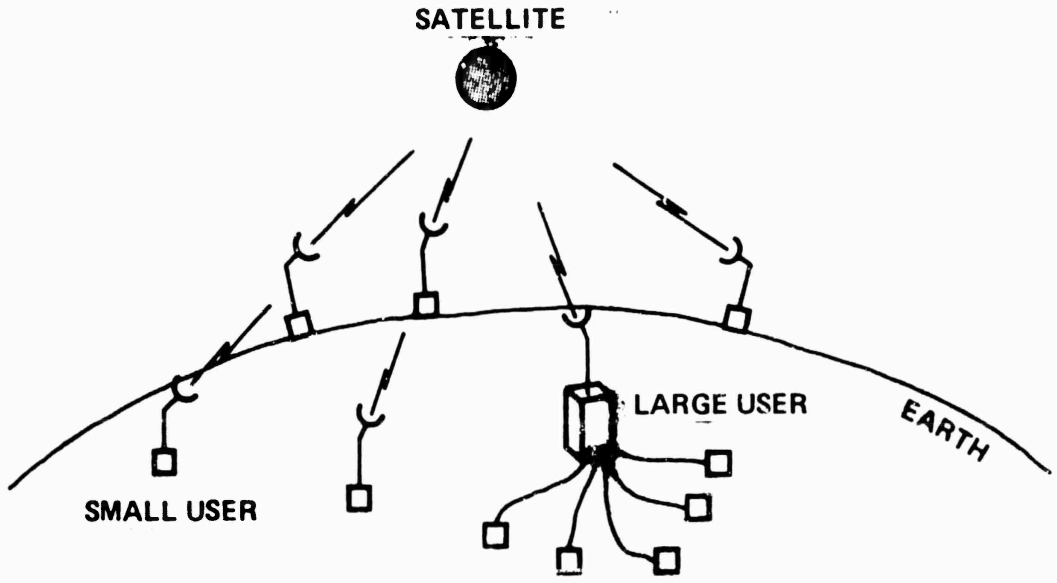


Figure 3-8. The Large User Model.

3.3.2 Throughput-Delay Results

We consider a channel user population consisting of a single large user with buffering and scheduling capabilities as described in Section 2.3.2, and a population of small users as in the infinite population model. Hence, we distinguish corresponding to the large user and the smaller users two channel input sources, both of which are assumed to be independent processes with stationary Poisson distributions. The (combined) input source to the small users

is at a rate of S_1 packets/slot. The input source to the large user is at a rate of S_2 packets/slot. The channel input rate is then given by

$$S = S_1 + S_2$$

The channel traffic in a time slot consists of packet transmissions by both the small users and large user. The large user resolves any conflict among its own packets competing for transmission in a time slot. The highest priority packet is transmitted and the rest of the competing packets are rescheduled for a later time. We define station traffic to be a random variable representing the number of packets in a time slot vying for transmission (i.e., for the transmitter) at the large user. The average station traffic is defined to be G_s packets/slot. Uniform randomization is assumed for both retransmitting packets which had a channel collision and rescheduling packets at the large user. Both station traffic and the portion of channel traffic due to the small users are assumed to be independent processes, Poisson distributed and have stationary distributions (within the time period of interest). As in the infinite population model, these assumptions represent approximations in our analytic model and will be examined by simulations.

We let G be the channel traffic rate such that

$$G = G_1 + G_2$$

where G_1 is the traffic rate due to the small users and G_2 is the traffic rate due to the large user. Since we assume that the large

user attempts no transmission in a time slot if no packet is scheduled for then (although there may be packets scheduled for a later time) and always transmits if one or more packets are scheduled for the time slot, G_2 can be interpreted as the probability that station traffic is greater than or equal to one. We must have $0 \leq G_2 \leq 1$.

We shall solve for equilibrim solutions such that the throughput rates for the small users and the large user are equal to their respective input rates S_1 and S_2. The analysis is similar to the first order approximation analysis for the infinite population model such that the effects of the uniform randomization intervals K and L are included in our model. The analytic results are summarized below. Details of the analysis are presented in Appendix B.

Similar to Eq. (3.9), equilibrium channel input rates and traffic rates are related by the following equations:

$$S_1 = G_1 \frac{q_{1t}}{q_{1t} + 1 - q_{1n}} \tag{3.16}$$

and

$$S_2 = G_2 \frac{q_{2t}}{q_{2t} + 1 - q_{2n}} \tag{3.17}$$

where q_{in} and q_{it} $(i = 1, 2)$ are the probability of success for the transmission of a new packet or a previously collided packet respectively. Note that variables indexed by 1 refer to the small users and variables indexed by 2 refer to the large user. The complete set of nonlinear implicit equations involving S_i, G_i, q_{in} and q_{it} are derived and presented in Appendix B. These equations have been solved numerically and the results are given below.

The average packet delay for the two classes of users are given by

$$D_1 = R + 1 + E_1 \left[R + \frac{K+1}{2} \right] \tag{3.18}$$

$$D_2 = R + 1 + E_2 \left[R + \frac{K+1}{2} \right] + (E_n + E_2 E_t) \frac{L+1}{2} \tag{3.19}$$

where E_1 and E_2 are the average number of retransmissions per packet for the small users and large user respectively; E_n and E_t are the number of reschedules per packet transmission at the large user conditioning on a new packet and a previously collided packet respectively. Recall that the average retransmission delay is $R + \frac{K+1}{2}$ and the average reschedule delay is $\frac{L+1}{2}$.

Limiting results

In the limit as $K, L \to \infty$, it is shown in Appendix B that our first order approximation results reduce to explicit solutions which could have been derived by direct arguments using the zeroth order approximation approach. (These results correspond to infinite average packet delay.) We have in the limit as $K, L \to \infty$,

$$q_{1n} = q_{1t} = e^{-G_1} (1 - G_2) \tag{3.20}$$

$$S_1 = G_1 e^{-G_1} (1 - G_2) \tag{3.21}$$

$$q_{2n} = q_{2t} = e^{-G_1} \tag{3.22}$$

$$S_2 = G_2 e^{-G_1} \tag{3.23}$$

$$G_2 = 1 - e^{-G_s} \tag{3.24}$$

The limiting channel throughput rate is then

$$S = (G - G_1 G_2) e^{-G_1} \tag{3.25}$$

where we recall $S = S_1 + S_2$ and $G = G_1 + G_2$. From the last equation, it can easily be shown that given either S_1 or S_2, S is maximized if the condition

$$G = G_1 + G_2 = 1$$

is satisfied. This proof was first given by L. Roberts in an unpublished note and was later generalized by Abramson [ABRA 73] to various other channel user populations. Abramson's result will be discussed in the next section.

In Fig. 3-9, we show a qualitative diagram of the 3-dimensional surface for S as a function of G_1 and G_2 (for the limiting case K, L approaching infinity). Consider the following equations:

$$\frac{\partial S}{\partial G_2} = e^{-G_1} (1 - G_1)$$

$$\frac{\partial S}{\partial G_1} = -e^{-G_1} (G - G_1 G_2 - 1 + G_2)$$

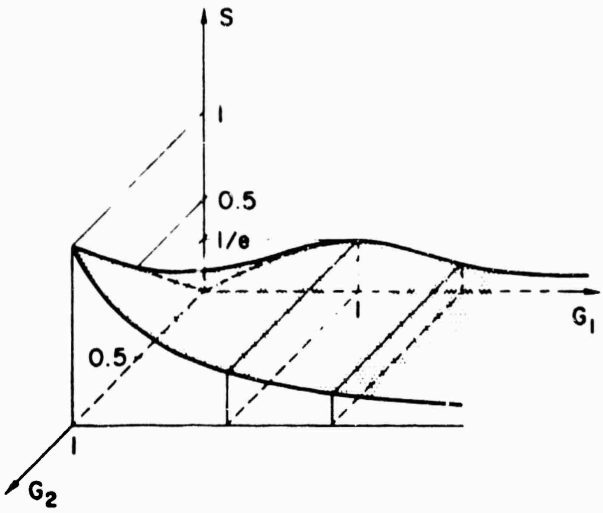


Figure 3-9. Throughput Surface.

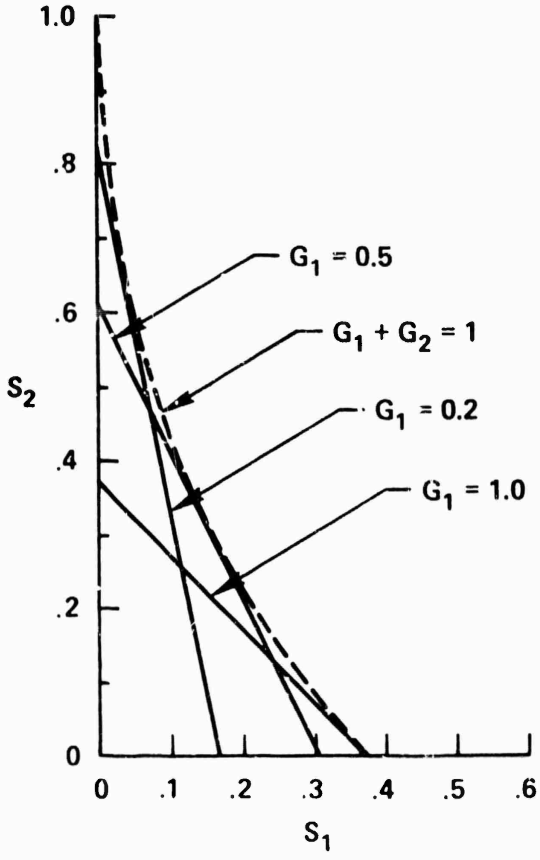


Figure 3-10. Allowable Throughput Rates for the Large User Model.

We see that for constant $G_1 < 1$, S increases linearly with G_2. For constant $G_1 > 1$, S decreases linearly as G_2 increases. In addition, for constant $G_2 < \frac{1}{2}$, S has a maximum value at $G_1 = (1 - 2G_2)/(1 - G_2)$, and for constant $G_2 > \frac{1}{2}$, S decreases as G_1 increases and the maximum throughput occurs at $S = G_2$ in the $G_1 = 0$ plane.

Numerical results

The maximum throughput contour given by letting $G = G_1 + G_2 = 1$ is shown in Fig. 3-10 along with throughput contours at constant G_1. We note in these last two figures that a channel throughput rate equal to 1 is achievable whenever G_1 (and hence, the throughput rate S_1 of the small users) drops to zero, in which case $S = S_2 = G = 1$; this then corresponds to the use of a dedicated channel.

We next present numerical results on throughput-delay tradeoffs for the finite K case; in all of these computations, we let $L = K$, thereby eliminating one parameter. In Fig. 3-11, we show the tradeoff between channel throughput rate and average packet delay for $S_1 = 0.1$, where the average packet delay D is defined to be $(S_1D_1 + S_2D_2)/S$. We show in this figure the equilibrium contours of D at constant values of K. The optimum performance envelope is given. Also shown are optimum performance envelopes for D_1 and D_2. We see that if we are willing to reduce the throughput of the small users from its maximum of $S_1 = 0.368$ to $S_1 = 0.1$, then we can drive the total throughput up to approximately $S = 0.52$ by introducing additional traffic from the large user. Note that the D_1 envelope

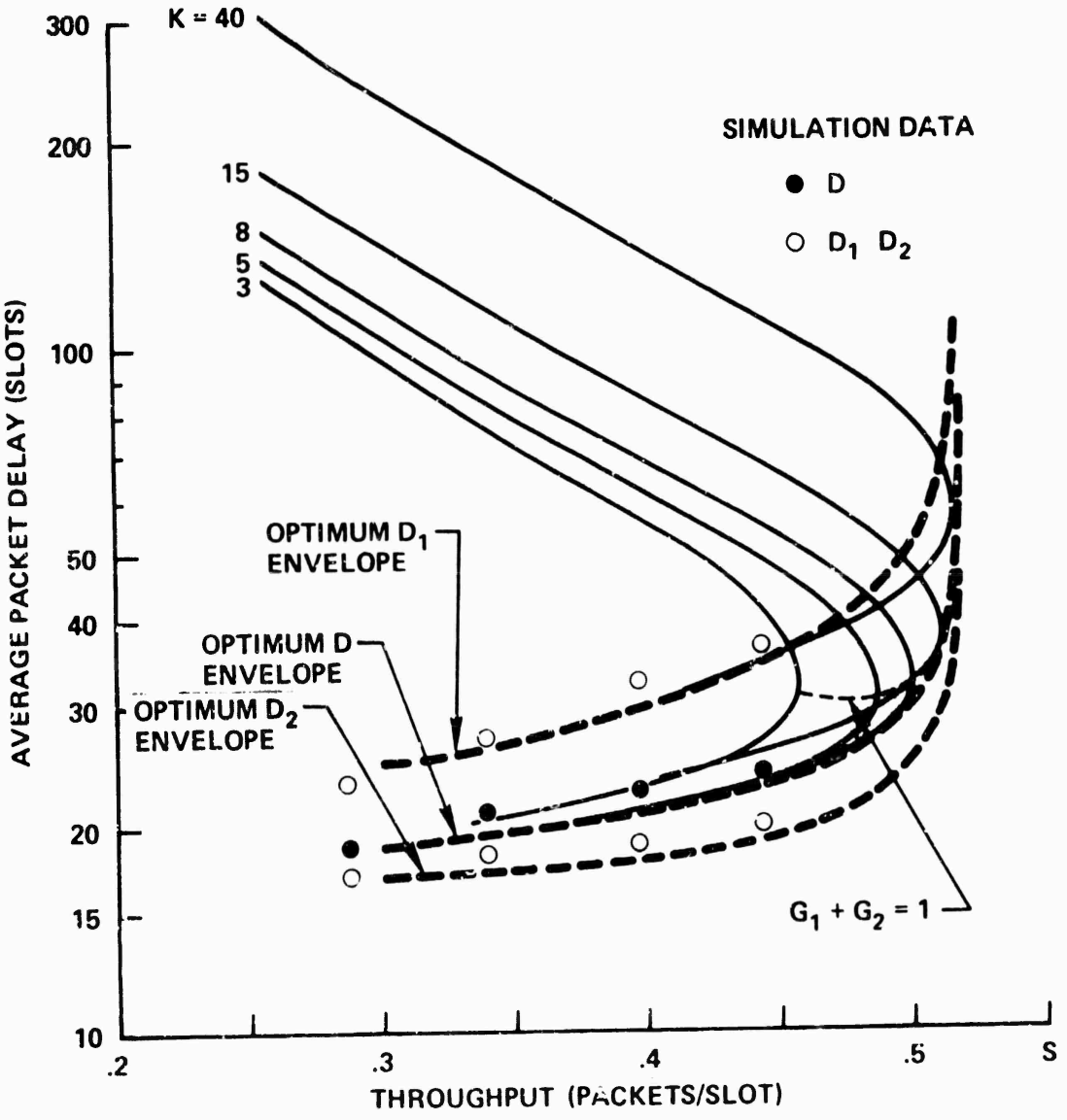


Figure 3-11. Throughput-Delay Tradeoff at $S_1 = 0.1$

is much higher than the D_2 envelope. Thus, our net gain in channel throughput is also at the expense of long delays for the small users. Once again, we note the sharp rise in average packet delay when S approaches the channel capacity.

In Fig. 3-12, we display a family of optimum throughput-delay performance envelopes for the large user model at fixed values of S_1 bounded by the optimum performance envelope of an infinite population model and that of a dedicated channel (modeled as a M/D/1 queue [KLEI 74D]). Note that as we reduce the background traffic, the system capacity increases slowly; however, when S_1 falls below 0.1, we begin to pick up significant gains. Also observe that each curve "peels off" from the infinite population model envelope at a value of $S = S_1$. The M/D/1 queue performance curve represents the absolute optimum performance contour for any method of using the channel when the channel input is Poisson; for input sources characterized by other probability distributions, we may use the G/D/1 queueing results to compute this absolute optimum performance contour.

Simulations

A simulation program was developed for the large user model. As in the infinite population model simulations, we found that the assumption of channel equilibrium is valid for the duration of a simulation run if a sufficiently small value of S or large value of K (and L) is used. Simulation points are indicated in Figs. 3-11 and 3-12 for those simulation runs which satisfied our channel equilibrium criterion (described earlier). The duration of each run was 5000 slots. Note that the analytic and simulation results agree very

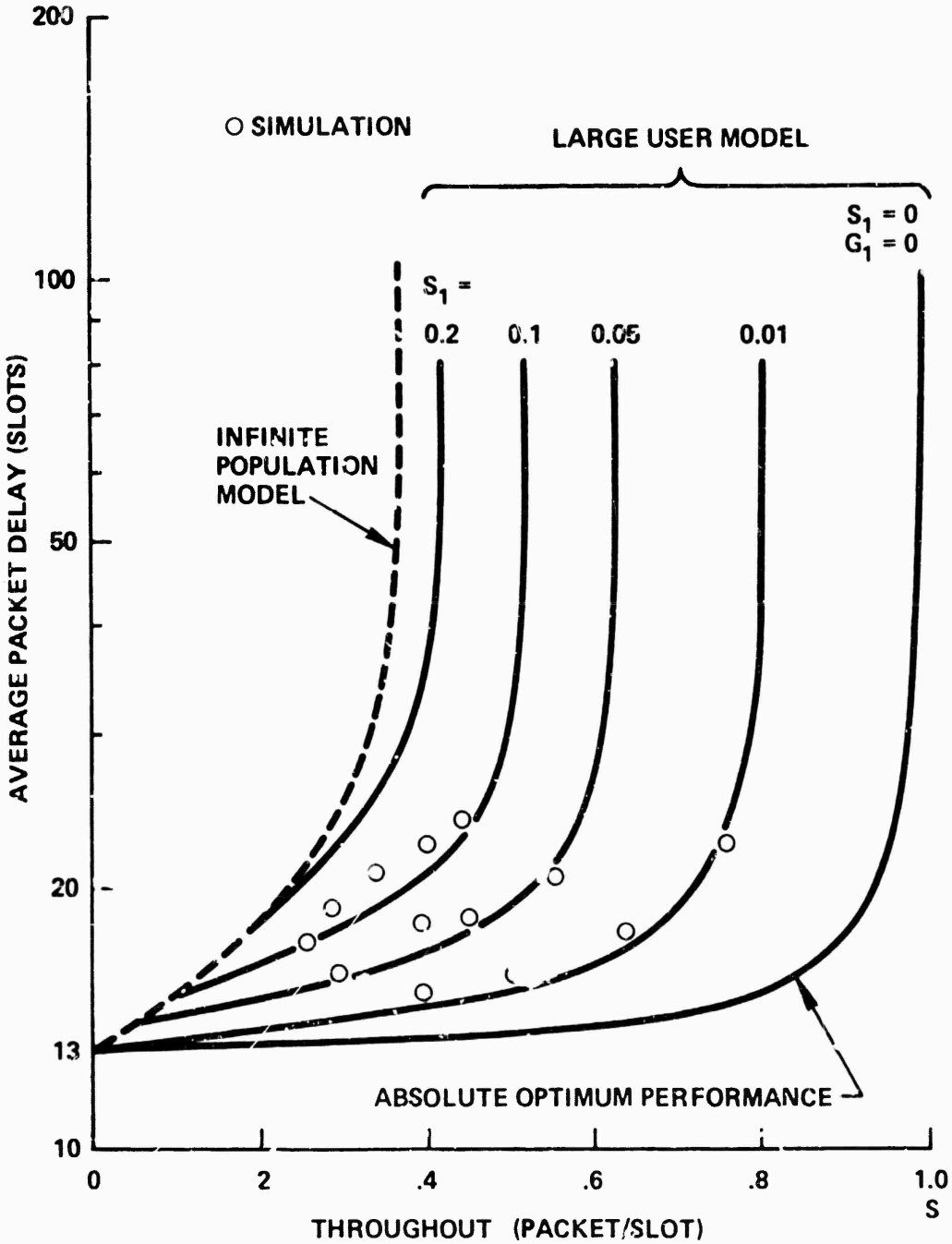


Figure 3-12. Optimum Throughput-Delay Tradeoffs.

well, thus justifying our analytic approximations. The channel input rates used in the simulations are much below the channel capacity; larger input rates can be used only with a very large K resulting in average delays much above the optimum performance envelopes.

3.4 The Finite Population Model

So far, we have considered the slotted ALOHA channel with a user population consisting of many small users modeled by a Poisson channel input. We have seen that by adding a large user with buffering and scheduling capabilities, the channel performance can be markedly improved if a significant portion of the channel input is due to the large user. In a real system, the input of this large user may change as time progresses. Moreover, the channel user population may include more than one large user. In this case, the first order approximation approach can still be applied to solve for the throughput-delay results. However, the large number of nonlinear implicit equations that must be solved numerically renders this approach computationally unattractive. In this section, the much simpler zeroth order approximation approach is adopted and some general results are presented on the channel capacity of the finite population model. Throughput-delay tradeoffs will then be examined by simulations.

3.4.1 Channel Capacity

The results in this section were first obtained by Abramson [ABRA 73].

Given M large users with channel input rates $S_1, S_2, \ldots, S_M$ and traffic rates $G_1, G_2, \ldots, G_M$. Note that G_i corresponds to the probability of the i^{th} user transmitting in a time slot (i.e.,

the probability of having one or more packets scheduled for transmission in that time slot as discussed previously in the large user model). The equilibrium values of S_i and G_i are related by

$$S_i = G_i \prod_{j=1, j\neq i}^{M} (1 - G_j) \qquad i = 1, 2, \ldots, M \qquad (3.26)$$

For any set of M acceptable traffic rates $G_1, G_2, \ldots, G_M$, these M equations define a set of input (throughput) rates $S_1, S_2, \ldots, S_M$ corresponding to a region in the M-dimensional space whose coordinates are the S_i. In order to find the boundary of this region, we calculate the Jacobian,*

$$J\left(\frac{S_1, S_2, \ldots, S_M}{G_1, G_2, \ldots, G_M}\right)$$

Since

$$\frac{\partial S_j}{\partial G_k} = \begin{cases} \prod_{\substack{i=1 \\ i\neq j}}^{M} (1 - G_i) & j = k \\ -G_j \prod_{\substack{i=1 \\ i\neq j,k}}^{M} (1 - G_i) & j \neq k \end{cases}$$

*This is the determinant of a M x M matrix whose jk^{th} element is $\frac{\partial S_j}{\partial G_k}$.

the Jacobian can be written as

$$J\left(\frac{S_1, S_2, \ldots, S_M}{G_1, G_2, \ldots, G_M}\right) = \alpha^{M-2} \begin{vmatrix} (1 - G_1) & -G_1 & -G_1 & \\ -G_2 & (1 - G_2) & -G_2 & \cdots \\ -G_3 & -G_3 & (1 - G_3) & \\ & \vdots & & \end{vmatrix}$$

$$= \alpha^{M-2} (1 - G_1 - G_2 - \ldots - G_M) \qquad (3.27)$$

where $\alpha = \prod_{j=1}^{M} (1 - G_j)$.

Equating the Jacobian to zero,* the boundary of the M-dimensional region of allowable input rates is defined by the condition

$$\sum_{i=1}^{M} G_i = 1 \qquad (3.28)$$

Examples

Consider two groups of users with M_1 users in group 1 and M_2 users in group 2 and let $M = M_1 + M_2$. Suppose $\frac{S_1}{M_1}$ and $\frac{G_1}{M_1}$ are the input and traffic rates of each user in group 1, and $\frac{S_2}{M_1}$ and $\frac{G_2}{M_2}$ are

---

* See Section 3.2 of [BEVE 70].

the input and traffic rates of each user in group 2. In this case, the M equations in Eqs. (3.26) become the two equations

$$S_1 = G_1\left(1 - \frac{G_1}{M_1}\right)^{M_1-1}\left(1 - \frac{G_2}{M_2}\right)^{M_2}$$

$$S_2 = G_2\left(1 - \frac{G_2}{M_2}\right)^{M_2-1}\left(1 - \frac{G_1}{M_1}\right)^{M_1} \qquad (3.29)$$

which map the region of acceptable traffic rates in the (G_1, G_2) plane into the region of allowable input rates in the (S_1, S_2) plane, the boundary of which is defined by the condition

$$G_1 + G_2 = 1 \qquad (3.30)$$

Substituting Eq. (3.30) into Eqs. (3.29), the resulting equation can be solved numerically for the maximum throughput contour (i.e., boundary of the allowable region of input rates) in the (S_1, S_2) plane. Several examples of such maximum throughput contours are shown in Fig. 3-13. Note that the special cases $(M_1, M_2) = (\infty, 1)$ and (∞, ∞) correspond to the large user model and the infinite population model respectively.

3.4.2 Simulation Results

A simulation program was developed for the finite population model. As in previous simulations for the infinite population model and the large user model, the assumption of channel equilibrium is valid for the duration of a simulation run if sufficiently small

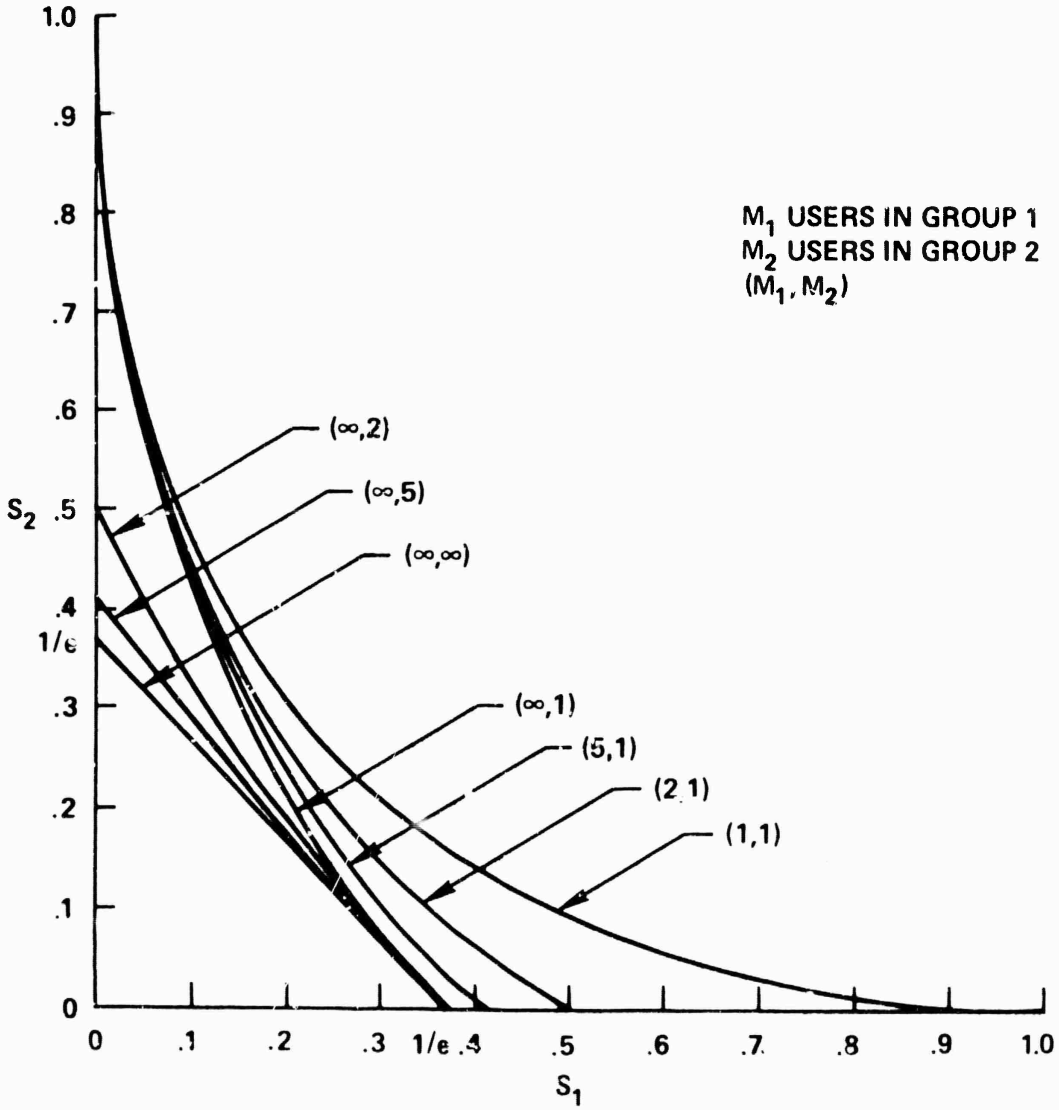


Figure 3-13. Allowable Throughput Rates for the Finite Population Model.

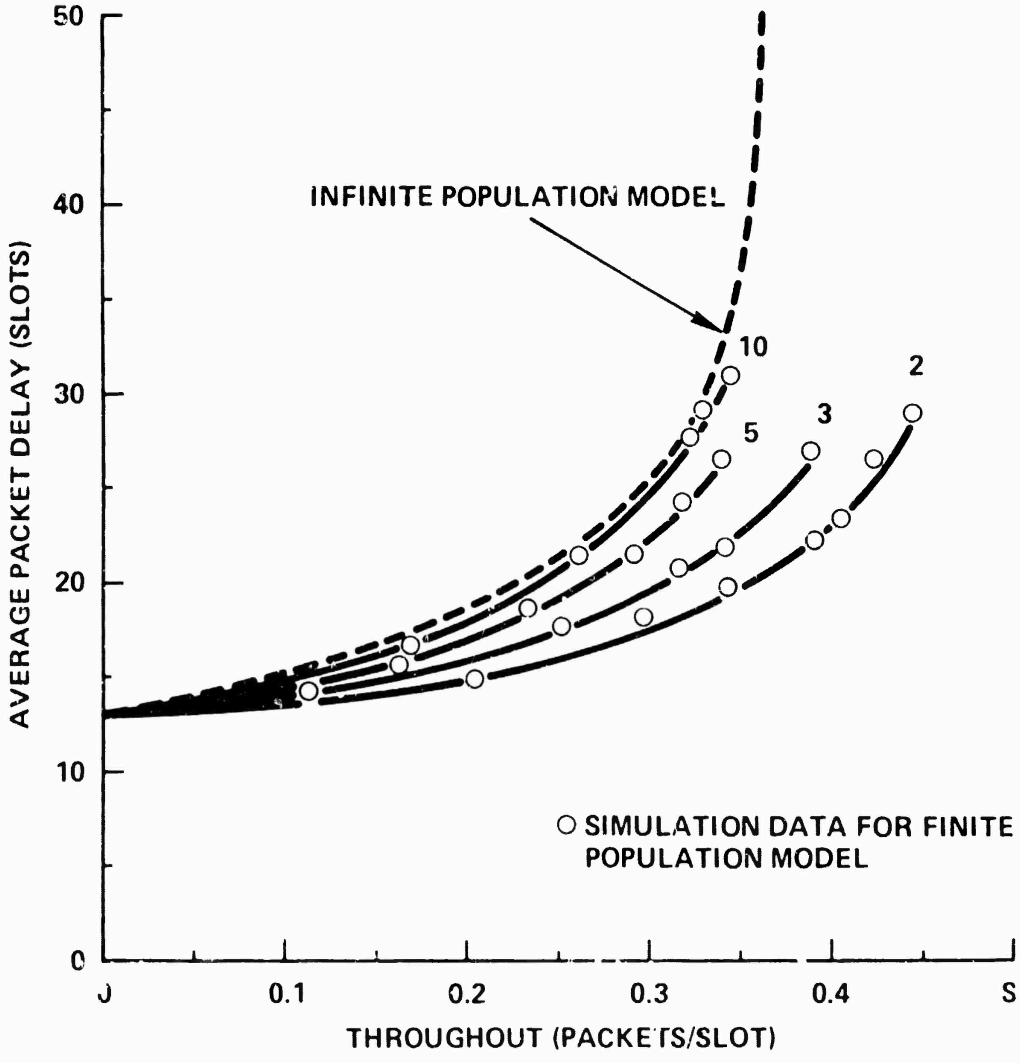


Figure 3-1. Throughput-Delay Tradeoffs for the Finite Population Model.

values of S_i or large values of K (and L) are used. We show in Fig. 3-14 throughput-delay tradeoff performances for the finite population model consisting of 2, 3, 5 and 10 large users; in each case, the channel input rate is assumed to be equally divided among the users. The infinite population model optimum envelope is also shown for comparison. Note that when the channel user population has 10 large users, the large user effect disappears and the throughput-delay results already approximate closely those of an infinite user population.

CHAPTER 4

CHANNEL DYNAMICS

In the last chapter, analytic models were developed to predict the equilibrium throughput-delay performance of the slotted ALOHA channel under various assumptions. Many of these assumptions (e.g., the independence assumption, the Poisson assumption and the stationarity assumption) represent merely approximations to the physical situation. However, without them the mathematical analysis becomes very complex and solutions are difficult to come by. The source of difficulty lies in the dimensionality of the state vector. (The state vector of a system consists of all the variables of interest such that knowledge of them at time t_1 and knowledge of all system inputs in the interval $[t_1,t_2]$ are sufficient to determine uniquely the state vector at time $t_2 > t_1$.) For the channel model under consideration, we must include in the state vector, channel information for as many time slots as the maximum value of a retransmission delay. Furthermore, each component of the state vector may take on a large number (possibly infinite) of values.

In this chapter, we first formulate a Markov chain model with none of the assumptions mentioned above and obtain a recursive transform equation which characterizes the time behavior of the channel. However, no simple solution to the transform equation has been obtained. Such an exercise in symbol manipulations serves only to illustrate the difficulty and futility of an exact mathematical analysis. Next, we adopt a weakened version of the independence assumption for channel traffic and show, for the infinite population model, that as the uniform

randomization interval K approaches infinity, the channel traffic is Poisson distributed. At the same time, the average channel traffic as a function of time is given by a difference equation. This equation permits us to investigate the dynamic behavior of the channel subject to a time varying input. Since only expected values are involved, the difference equation represents a deterministic approximation or fluid approximation [KLEI 74D, NEWE 68] of the original stochastic process. Similar dynamic channel behavior was predicted by Rettberg [RETT 72].

4.1 An Exact Analysis

We shall analyze the slotted ALOHA channel described in Section 2.3 without most of the assumptions made in the last chapter. As before, V^t and X^t are random variables representing the channel input and channel traffic in the t^{th} time slot. The only assumption we shall need in this section is that V^t is an independent process and independent of the channel state. The channel state vector at time t is given by the set of $R + K$ variables $\{X^t, X^{t-1}, \ldots X^{t-R-K+1}\}$. (Note that $R + K$ is the maximum value of a retransmission delay.) We define the channel state vector

$$\underline{X}^t = \begin{bmatrix} X_1^t \\ X_2^t \\ \vdots \\ X_{R+K}^t \end{bmatrix} = \begin{bmatrix} X^t \\ X^{t-1} \\ \vdots \\ X^{t-R-K+1} \end{bmatrix}$$

which is a random vector with probability distribution

$$P^t(\underline{x}) = \text{Prob}\,[\underline{\lambda}^t = \underline{x}] \qquad \underline{x} \in S$$

where S is the admissible state space and $\underline{x}$ is an integer-valued $(R + K)$-dimensional vector in S. $\underline{X}^t$ is a discrete state discrete time Markov chain which will be completely specified by its one-step state transition probabilities

$$P^t(\underline{y}|\underline{x}) = \text{Prob}[\underline{X}^{t+1} = \underline{y}|\underline{X}^t = \underline{x}] \qquad \underline{x},\underline{y} \in S$$

such that

$$P^{t+1}(\underline{y}) = \sum_{\underline{x}\in S} P^t(\underline{y}|\underline{x})P^t(\underline{x}) \qquad \underline{y} \in S \qquad (4.1)$$

We now define,

$$v_i^t = \text{Prob}[V^t = i] \qquad i = 0, 1, 2, \ldots$$

and

$$\lambda(m) = \begin{cases} 0 & m = 1 \\ m & m \neq 1 \end{cases}$$

The one-step state transition probabilities at time t for the Markov chain $\underline{X}^t$ are given below.

$$P^t(\underline{y}|\underline{x}) = \begin{cases} 0 & \text{if } y_i \neq x_{i-1} \\ & \forall\ i = 2, 3, \ldots, R+K \\ \sum\limits_{\substack{i=0 \\ i\le\ell}}^{y_1} v_{y_1-i}^{t+1} \binom{\ell}{i} \left(\frac{1}{K}\right)^i \left(1-\frac{1}{K}\right)^{\ell-i} & \text{otherwise} \end{cases} \qquad (4.2)$$

where

$$\ell = \sum_{j=1}^{K} \chi(x_{R+j})$$

is the total number of packets which collided in the K slots (from the $(t - R)^{th}$ to the $(t - R - K + 1)^{th}$) such as shown in Fig. 3.1. Note that each such packet retransmits into the $(t + 1)^{st}$ slot independently with probability $\frac{1}{K}$. Note also that for $i = 2, 3, \ldots, R + K$, the event $[y_i \neq x_{i-1}]$ is impossible and thus has zero probability (since both y_i and x_{i-1} represent the value of channel traffic in the same time slot).

Now given an initial probability distribution for the channel traffic in R consecutive time slots, the stochastic behavior of the channel as a function of time is predicted by Eqs. (4.1) and (4.2).

A recursive transform equation

We first define the following transforms,

$$V^t(z) = \sum_{i=0}^{\infty} z^i \, v_i^t$$

and

$$Q^t(\underline{z}) = Q^t(z_1, z_2, \ldots, z_{R+K})$$

$$= \sum_{x_1=0}^{\infty} \cdots \sum_{x_{R+K}=0}^{\infty} \left(\prod_{j=1}^{R+K} z_j^{x_j} \right) p^t(\underline{x})$$

From Eqs. (4.1) and (4.2), a recursive transform equation relating $Q^{t+1}(\underline{z})$ to $V^{t+1}(z)$ and $Q^t(\underline{z})$ can be derived (see Appendix C) and is

given below.

$$Q^{t+1}(\underline{z}) = V^{t+1}(z_1)\left[\sum_{\{(\varepsilon_1,\ldots,\varepsilon_K)\,|\,\varepsilon_j=0,1\}} Q^t(\underline{z};\underline{\varepsilon})\right] \qquad (4.3)$$

where $\underline{\varepsilon}$ is a K-dimensional vector and $Q^t(\underline{z};\underline{\varepsilon})$ for a given $\underline{\varepsilon}$ can be obtained from $Q^t(\underline{z})$ by the following algorithm:

(1) Initialize $Q^t(\underline{z};\underline{\varepsilon}) \leftarrow Q^t(z_2,\ldots,z_{R+1},y_1,\ldots,y_K)$ and $j \leftarrow 1$

(2) If $j = K$, go to (5)

(3) If $\varepsilon_j = 0$,

replace y_j by $z_{R+j+1}\left(1 - \frac{1}{K} + \frac{z_1}{K}\right)$ in $Q^t(\underline{z};\underline{\varepsilon})$,

else $Q^t(\underline{z};\underline{\varepsilon}) \leftarrow z_{R+j+1}\,\frac{(1-z_1)}{K} \cdot \frac{\partial}{\partial y_j} Q^t(\underline{z};\underline{\varepsilon})\Big|_{y_j = 0}$

(4) $j \leftarrow j + 1$ and go to (2)

(5) If $\varepsilon_K = 0$,

replace y_K by $\left(1 - \frac{1}{K} + \frac{z_1}{K}\right)$ in $Q^t(\underline{z};\underline{\varepsilon})$,

else $Q^t(\underline{z};\underline{\varepsilon}) \leftarrow \frac{1-z_1}{K} \cdot \frac{\partial}{\partial y_K} Q^t(\underline{z},\underline{\varepsilon})\Big|_{y_K = 0}$

The notation $A \leftarrow F(A)$ means: evaluate $F(A)$ which then becomes the new expression for A. As an example, for the case when $R = 2$ and $K = 2$, we get the following transform equation.

$$Q^{t+1}(z_1,z_2,z_3,z_4) = V^{t+1}(z_1)\Bigg[Q^t\left(z_2,z_3,z_4\left(1-\frac{1}{K}+\frac{z_1}{K}\right),\left(1-\frac{1}{K}+\frac{z_1}{K}\right)\right)$$
$$+\frac{z_4(1-z_1)}{K}\cdot\frac{\partial}{\partial y_1}Q^t\left(z_2,z_3,y_1,\left(1-\frac{1}{K}+\frac{z_1}{K}\right)\right)\Bigg|_{y_1=0}$$
$$+\frac{1-z_1}{K}\cdot\frac{\partial}{\partial y_2}Q^t\left(z_2,z_3,z_4\left(1-\frac{1}{K}+\frac{z_1}{K}\right),y_2\right)\Bigg|_{y_2=0}$$
$$+\frac{z_4(1-z_1)^2}{K^2}\frac{\partial^2}{\partial y_1\partial y_2}Q^t(z_2,z_3,y_1,y_2)\Bigg|_{y_1,y_2=0}\Bigg]$$

The above equation demonstrates the complexity of the transform equation even for small values of R and K. No solution to Eq. (4.3) has been found. The above analysis serves to illustrate the difficulty and futility of an exact mathematical analysis and motivates our use of approximations.

4.2 An Approximate Solution

In this section, we shall analyze the same model above with an additional assumption. We shall assume that the channel traffic within any K consecutive time slots are independent of each other* so that it suffices to solve for the probabilities,

$$P_i^t = \text{Prob}[X^t = i] \qquad i = 0, 1, 2, \ldots$$

and the transform,

$$Q^t(z) = \sum_{i=0}^{\infty} z^i P_i^t$$

*This is a weakened version of the independence assumption for channel traffic used in Chapter 3 and will be referred to as the weak independence assumption for channel traffic. Recall that the (strong) independence assumption gave very accurate results as verified by simulations in Chapter 3.

instead of $Q^t(z)$. We define the expected values of the random variables X^t and V^t as

$$G^t = E[X^t]$$

and

$$S^t = E[V^t]$$

Another transform equation

A transform equation similar to Eq. (4.3) can be derived under the weak independence assumption (see Appendix C) and is given below.

$$Q^t(z) = V^t(z) \prod_{j=1}^{K} \left[Q^{t-R-j} \left(1 - \frac{1}{K} + \frac{z}{K} \right) + P_1^{t-R-j} \frac{1-z}{K} \right] \qquad (4.4)$$

Eq. (4.4) can be solved recursively for $Q^t(z)$ given initial probability distributions of the channel traffic in $R + K$ consecutive time slots. Alternatively, $Q^t(z)$ can be approximated arbitrarily well by solving for P_1^t and a finite number of the moments of the channel traffic X^t . (Note that P_1^t represents the expected channel throughput in the t^{th} time slot.) By differentiating Eq. (4.4) with respect to z and setting z equal to zero, we obtained the following difference equation for G^t under our assumptions.

$$G^t = \frac{1}{K} \sum_{j=1}^{K} \left[G^{t-R-j} - P_1^{t-R-j} \right] + S^t \qquad (4.5)$$

Theorem 4.1 If the channel input is an independent Poisson process, then the channel traffic is Poisson distributed in the limit as $K \to \infty$ under the weak independence assumption, such that

$$Q^t(z) = e^{-G^t(1-z)} \tag{4.6}$$

and

$$P_1{}^t = G^t e^{-G^t} \tag{4.7}$$

where

$$G^t = \frac{1}{K} \sum_{j=1}^{K} \left(G^{t-R-j} - G^{t-R-j} \, e^{-G^{t-R-j}} \right) + S^t \tag{4.8}$$

Proof See Appendix C.

Equation (4.8) characterizes (approximately) the time behavior of the channel traffic subject to a time varying input when K is large. However, since only expected values are considered, this equation represents a fluid approximation of the stochastic process X^t. To incorporate statistical effects into the time behavior of the system, other techniques which account for some of the higher moments of X^t such as diffusion approximation [KLEI 74D, NEWE 68] may be employed.

Channel saturation described in the last chapter may arise as a result of either (a) statistical fluctuaions, or (b) time variations in the channel input. The effect of statistical fluctations will be studied in the next chapter. The effect of time varying inputs is examined below using Eq. (4.8).

4.3 Some Fluid Approximation Results

Given the Poisson channel input rate as a function of time, the expected channel traffic as a function of time can be obtained from the fluid approximation given by Eq. (4.8). In Figs. 4-1 and 4-2, we show the channel response to two input pulses. In both cases the channel input rate is initially equal to 0.3 packet/slot with the channel in equilibrium. The input rate is then increased to 0.8 packet/slot (well above the channel capacity of 0.368 packet/slot for an infinite population model) for 100 time slots. As a result, the channel traffic rate increases rapidly as the channel throughput rate decreases. The expected channel backlog (defined to be the net area between the channel input and throughput curves and corresponds to the expected total number of packets awaiting retransmission in all channel users) builds up. At the end of the 100 time slots, the channel input rate is reduced to 0.15 packet/slot in the first case. We see that the channel slowly returns to an equilibrium state (see Fig. 4-1). In the second case, the channel input rate is reduced at the end of the pulse to 0.25 packet/slot which, as we see in Fig. 4-2, is not small enough to prevent the channel from saturation. Simulations were performed for both cases using the simulation program developed for the infinite population model. The results are shown in Figs. 4-3 and 4-4. Note that each simulation point indicated actually represents an average value over a period of 50 time slots. Four simulations are shown for each of the two cases. We see that the fluid approximation results in Figs. 4-1 and 4-2 predict the

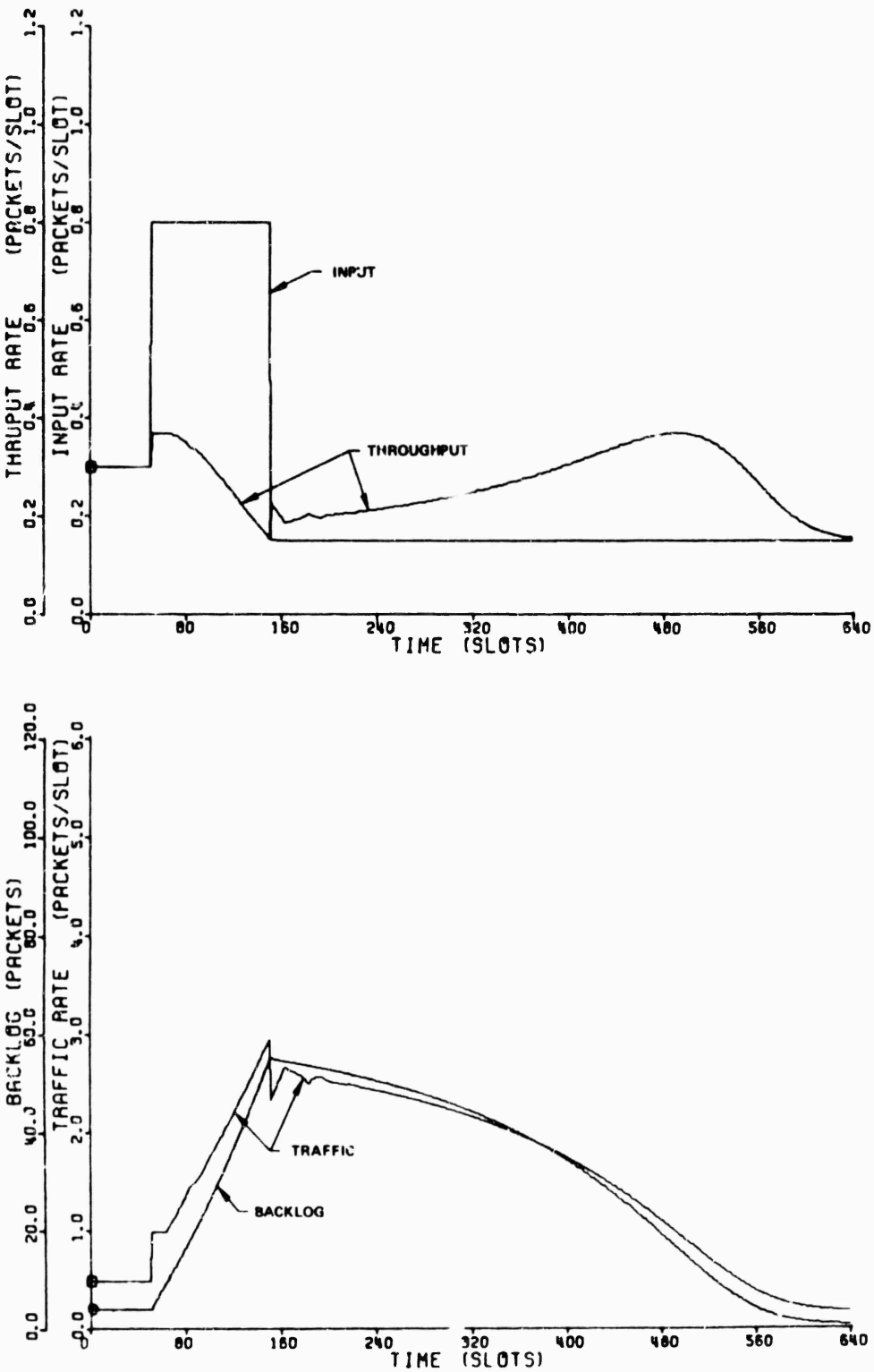


Figure 4-1. Channel Response to an Input Pulse (R = 12, K = 20).

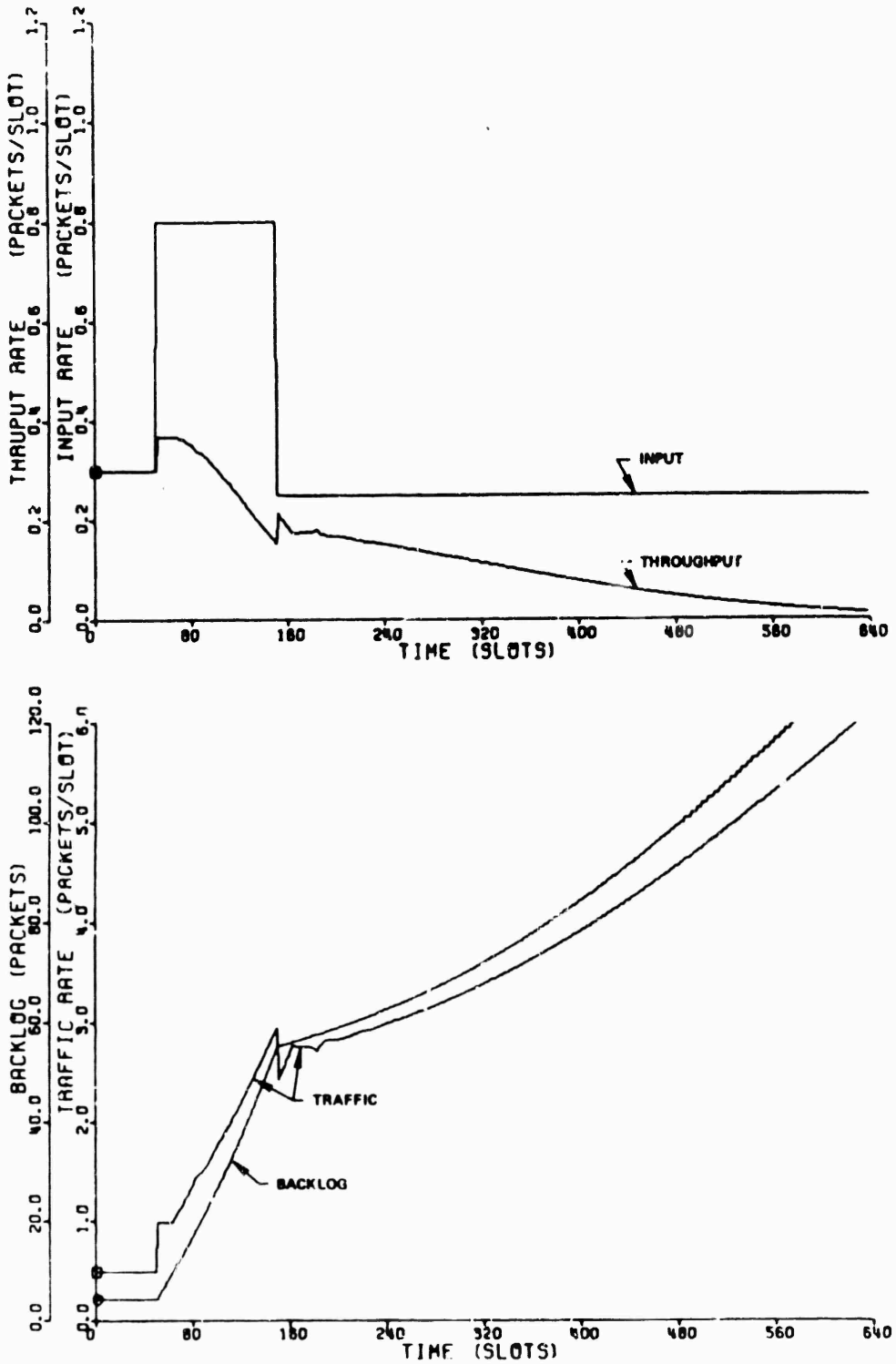


Figure 4-2. Channel Saturation (R = 12, K = 20)

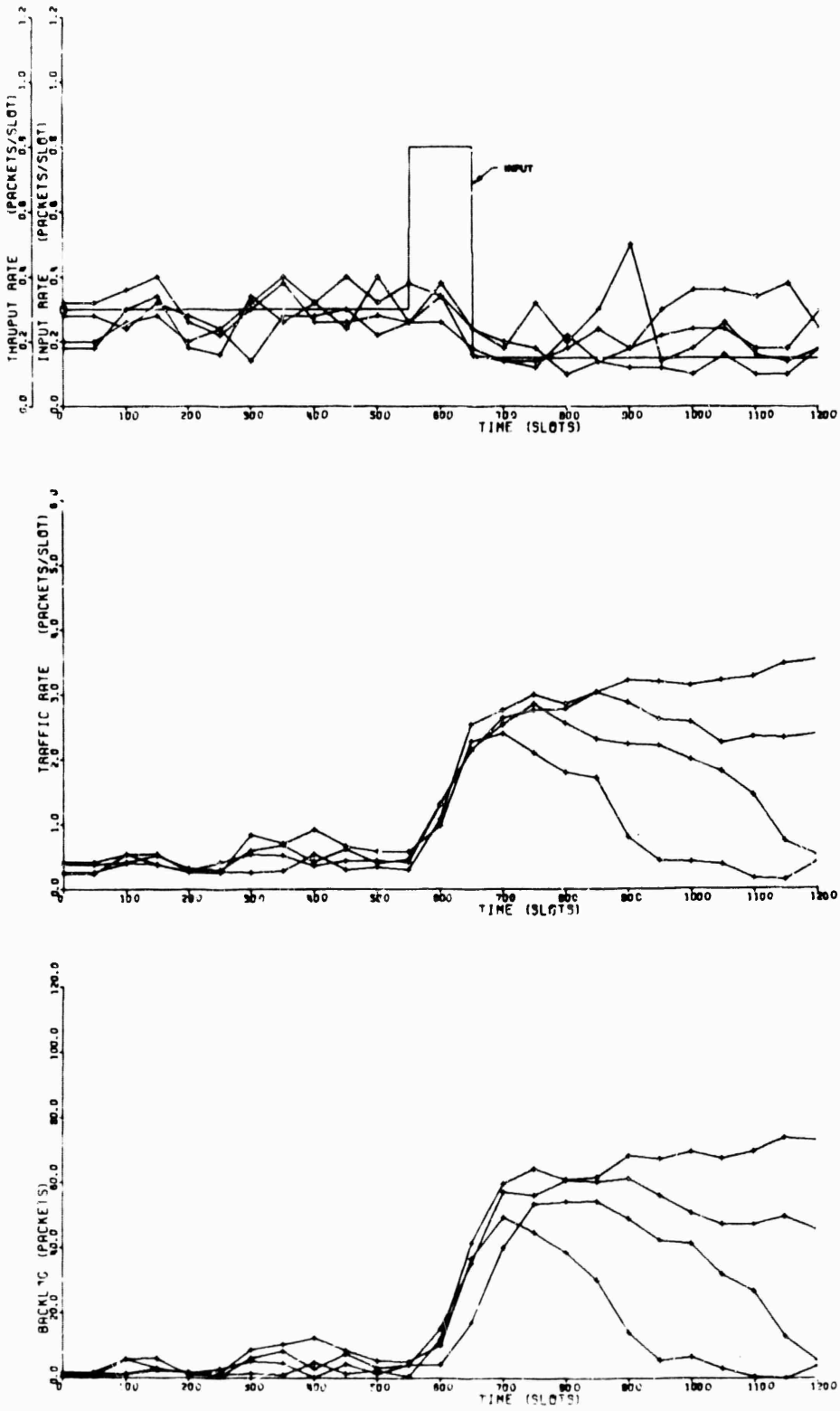


Figure 4-3. Simulations Corresponding to Figure 4-1.

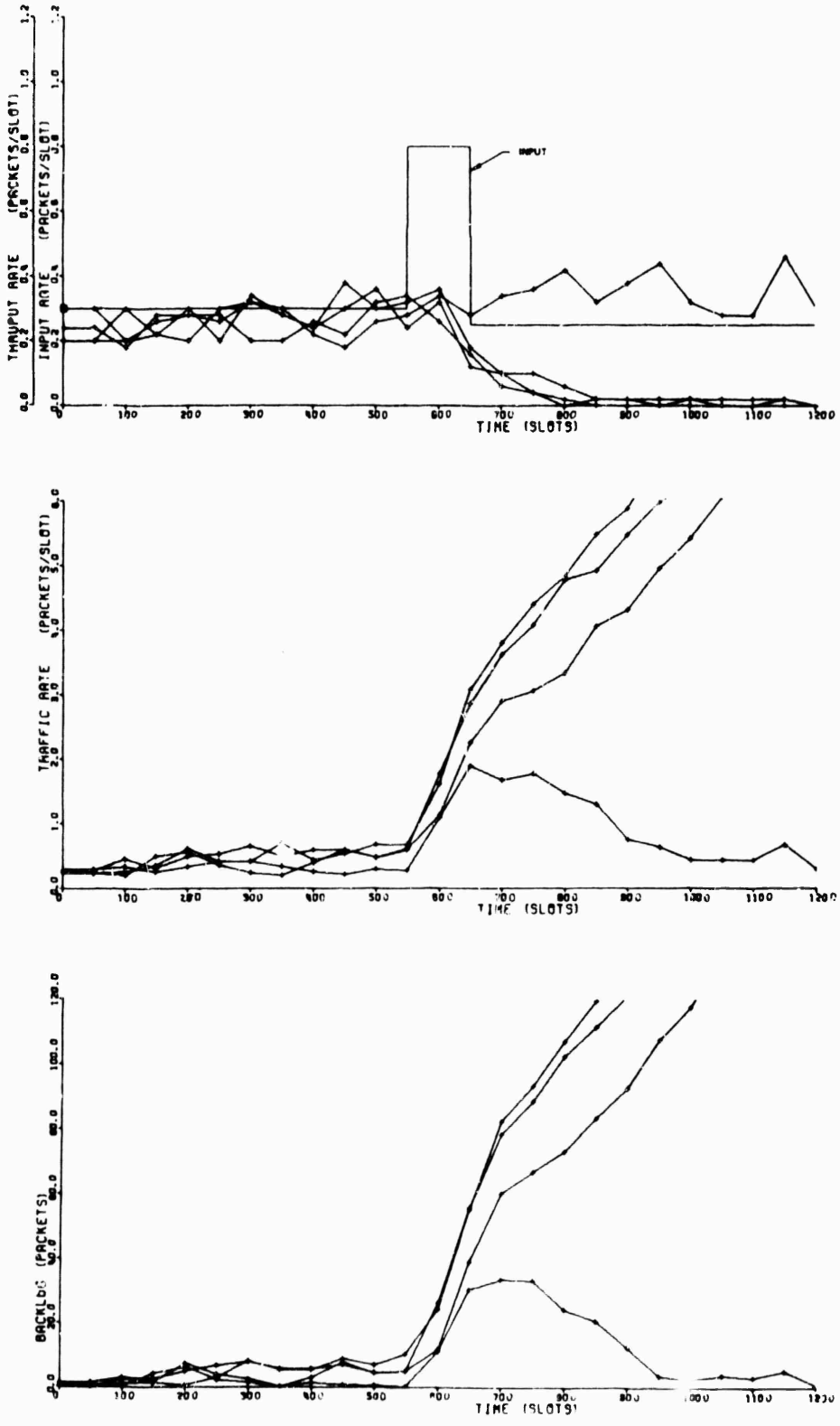


Figure 4-4. Simulations Corresponding to Figure 4-2.

general trend of the simulations. However, since the fluid approximation does not account for statistical fluctuations, there exist large variations among the simulation results.

In Fig. 4-5, we show, using Eq. (4.8), the channel response to a ramp pulse ("impulse") in the channel input rate. At the end of the pulse, the channel input rate is reduced to a small enough value so that the channel is able to return to an equilibrium state. Note that the channel has a natural frequency equal to the inverse of the expected retransmission delay (which is $R + (K + 1)/2$). (In Figs. 4-1 and 4-2 the channel oscillations are less pronounced as a result of a smaller input pulse and a larger K.)

In Fig. 4-6, we present results from the following experiment using Eq. (4.8). Starting with an equilibrium channel, an input pulse is applied until the expected channel backlog reaches some specified value B. The channel input rate is then reduced to some fixed value S'. The time the channel takes to return to an equilibrium state (the recovery time) is measured. (The criterion we adopt here for channel equilibrium is that the channel traffic rate must be less than one for $R + K$ consecutive time slots.) The experiment was carried out numerically using Eq. (4.8) for both rectangular and ramp pulses with different amplitudes. The initial equilibrium channel input rate $S_e = 0.2$ or 0.3 packet/slot. The channel recovery

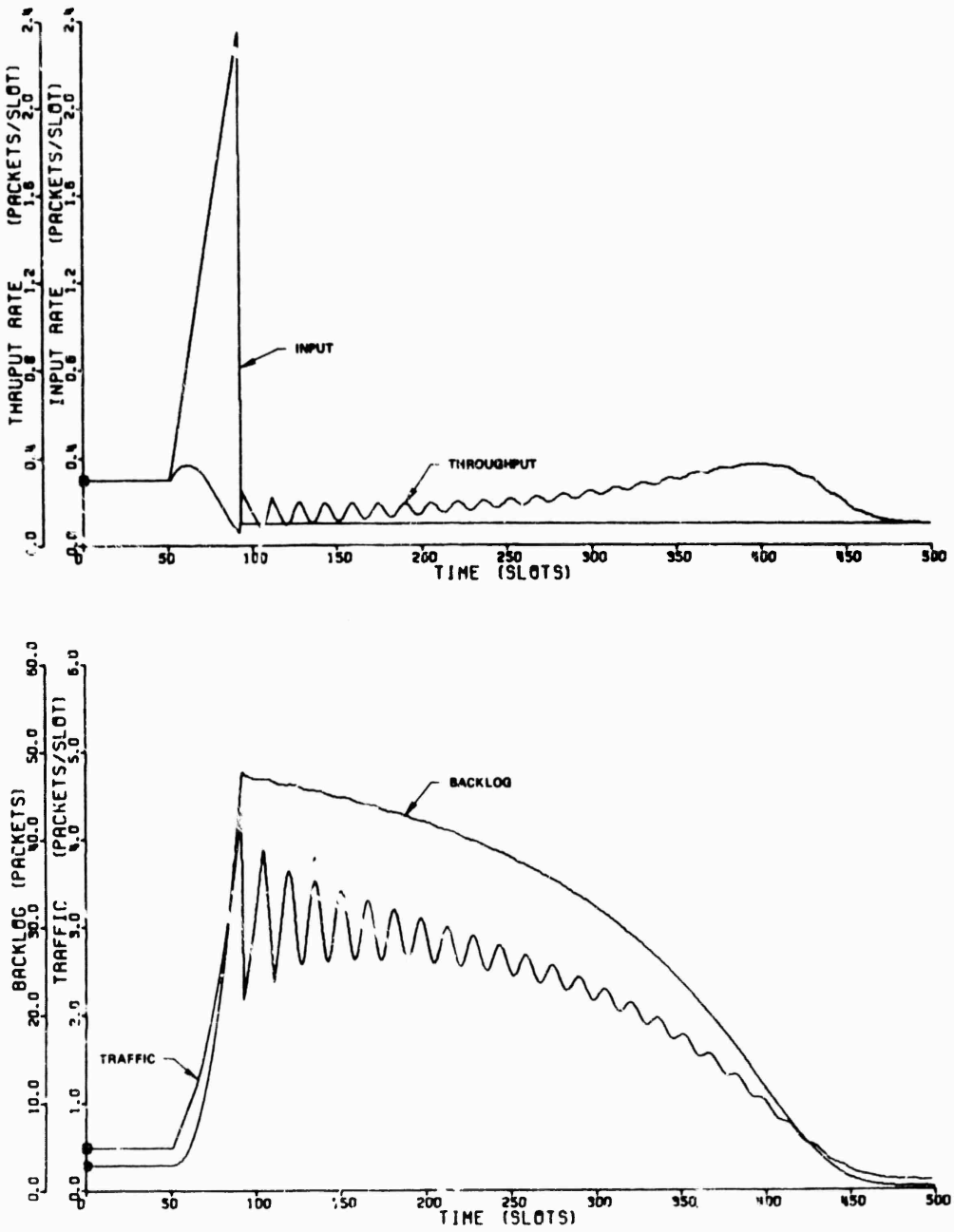


Figure 4-5. Channel Response to a Ramp Pulse (R = 12, K = 6).

time for the cases* considered is shown in Fig. 4-6 as a function of B for constant values of S'. Note that given S' there is a maximum value of B above which the channel recovery time is infinite, in which case a smaller S' must be used. It is interesting to note that the channel recovery time is insensitive to both the shape of the input pulse and S_e. The relevant variables are just B and S'. Recall that the expected channel backlog is the net area between the input and throughput curves. Thus, our results seem to indicate that the channel impulse response depends only upon the area under the impulse but not its shape, which reminds us of the response of linear systems [SCHW 65]! These results also suggest that instead of defining a complex state description such as in the previous sections, the channel behavior may be characterized quite adequately using the channel backlog size alone as a state variable.

---

*Four cases are considered:

(1) rectangular pulse, peak value = 2.35, S_e = 0.3

(2) rectangular pulse, peak value = 2.35, S_e = 0.2

(3) rectangular pulse, peak value = 1.35, S_e = 0.3

(4) ramp pulse, $S(t) = 0.35 + 0.05t$, $S_e = 0.3$

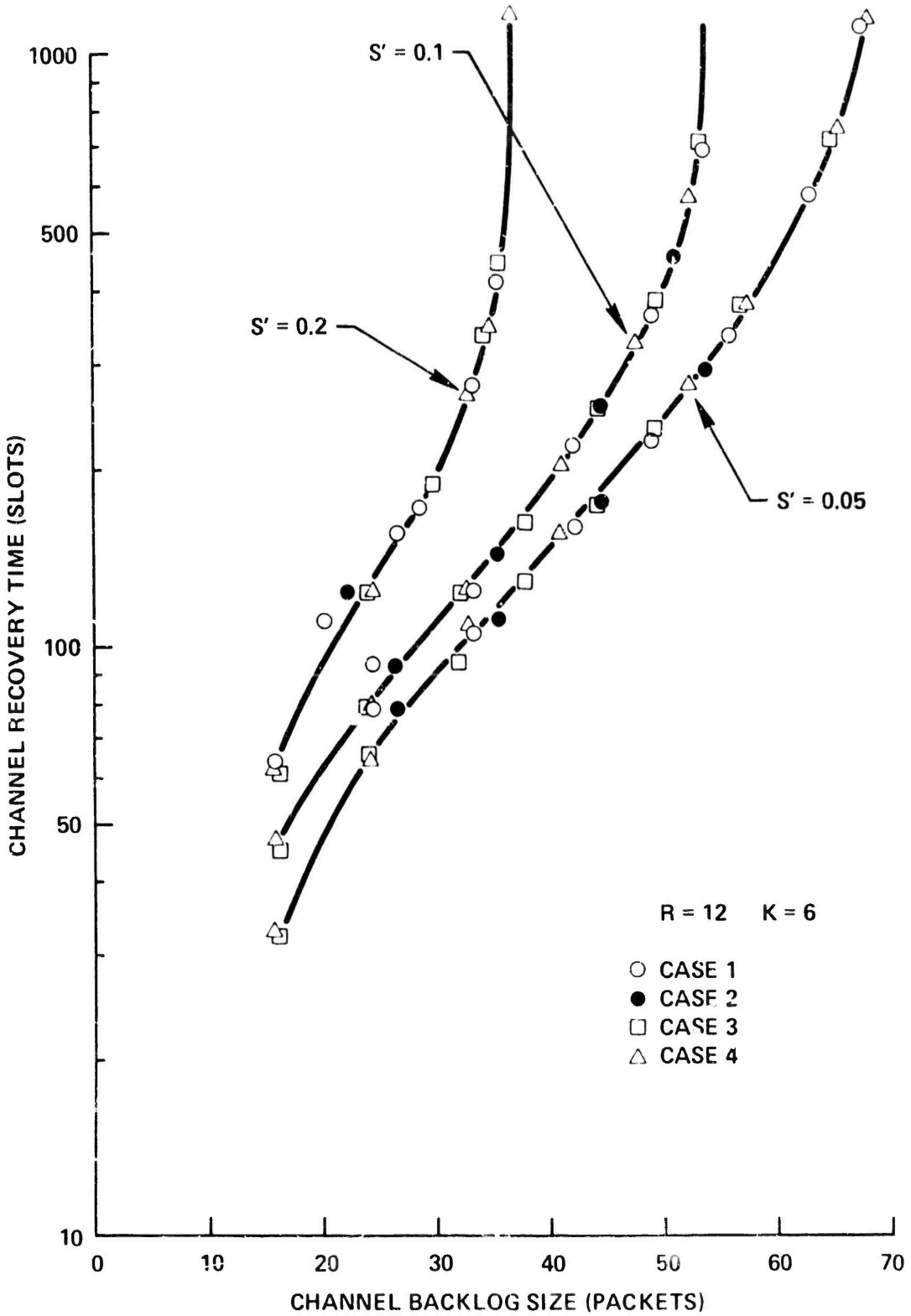


Figure 4-6. Channel Recovery Time Versus Channel Backlog Size (R = 12, K = 6)

CHAPTER 5

CHANNEL STABILITY

The slotted ALOHA random access method enables a multi-access channel to be statistically multiplexed in an efficient way by a large number of users. Such a system was studied in Chapter 3 as an infinite population model; equilibrium results on the channel throughput-delay performance were given. However, simulations have shown that the assumption of channel equilibrium may not always be valid. In fact, the channel, after some finite time period of quasi-stationary conditions, will drift into saturation with probability one. Thus, we realize that the equilibrium throughput-delay results are not sufficient to characterize the performance of the infinite population model. A more representative measure of channel performance in this case is the stability-throughput-delay tradeoff. To do so, we must first define channel stability and a stability measure.

We consider in this chapter a slotted ALOHA channel supporting a total of M users. The variable M is assumed to be large, but finite. We show below that the exact value of M is an important stability parameter. The purpose of this chapter is to characterize the instability phenomenon in the following ways:

- We define stable and unstable channels

- We show that in a stable channel, equilibrium throughput-delay results presented in Chapter 3 are achievable over an infinite time horizon. In an unstable channel, such channel performance is achievable only for some finite time period before the channel goes into saturation
- For an unstable channel, we define a stability measure and give an efficient computational procedure for its calculation
- Using the above stability measure, we examine the stability-throughput-delay tradeoff for an unstable channel

5.1 The Model

In the last chapter, we realized that the source of our difficulty in analysis lies in the complexity of the state description. Below we first define a mathematical model which characterizes the channel state by a single variable. Practical considerations and the model approximations to a physical system will then be examined. This mathematical model will also be adopted in the next chapter.

Our model is similar to the linear feedback model studied by Metcalfe who gave a steady-state analysis of the system behavior [METC 73A]. Lu [LU 73] studied the same model and characterized the time-dependent channel behavior through a set of linear difference equations. However, his approach (like our results in Section 4.1) cannot be easily applied to a system with many states (i.e., channel users).

5.1.1 The Mathematical Model

We consider a slotted ALOHA channel with a user population consisting of M small users (see Section 2.3). Each such user can be in one of two states: blocked or thinking. In the thinking state, a small user generates and transmits a new packet in a time slot with probability σ. A packet which had a channel collision and is waiting for retransmission is said to be backlogged. A small user with a backlogged packet is blocked in the sense that he cannot generate (or accept from his input source) a new packet for transmission. The retransmission delay RD of each backlogged packet is assumed to be geometrically distributed, i.e., each backlogged packet retransmits* in the current time slot with probability p.

Let N^t be a random variable (called channel backlog) representing the total number of backlogged packets at time t. Given that $N^t = n$, the channel input rate at time t is $S^t = (M - n)\sigma$. Note that S^t decreases linearly as n increases. Thus, this will also be referred to as the linear feedback model. The vector (N^t, S^t) will be denoted as the channel state vector. In this context, both M and σ may be functions of time. We shall assume M and σ to be time-invariant unless stated otherwise. In this case, N^t is a Markov process (chain) with stationary transition probabilities and serves as the state description for the system. The state space will now consist of the set of integers $\{0, 1, 2, \ldots, M\}$. The one-step state transition probabilities of N^t are, for $i = 0, 1, 2, \ldots$

---

*Assuming bursty users, we must have $p >> \sigma$.

$$p_{ij} = \text{Prob}[N^{t+1} = j \mid N^t = i]$$

$$= \begin{cases} 0 & j \le i - 2 \\ ip(1 - p)^{i-1}(1 - \sigma)^{M-i} & j = i - 1 \\ (1 - p)^{i}(M - i)\sigma(1 - \sigma)^{M-i-1} & \\ \quad + [1 - ip(1 - p)^{i-1}](1 - \sigma)^{M-i} & j = i \\ (M - i)\sigma(1 - \sigma)^{M-i-1}[1 - (1 - p)^{i}] & j = i + 1 \\ \binom{M - i}{j - i}\sigma^{j-i}(1 - \sigma)^{M-j} & j \ge i + 2 \end{cases} \tag{5.1}$$

For the infinite population model in which $M \to \infty$ and $\sigma \to 0$ such that $M\sigma = S$ which is constant and finite, the above equation becomes

$$p_{ij} = \begin{cases} 0 & j \le i - 2 \\ ip(1 - p)^{i-1} e^{-S} & j = i - 1 \\ (1 - p)^{i} Se^{-S} + [1 - ip(1 - p)^{i-1}]e^{-S} & j = i \\ Se^{-S}[1 - (1 - p)^{i}] & j = i + 1 \\ \frac{S^{j-i}}{(j - i)!} e^{-S} & j \ge i + 2 \end{cases} \tag{5.2}$$

5.1.2 Practical Considerations

The above mathematical model approximates a physical system in several ways. First, M and σ will be assumed to be time-

invariant. However, if we distinguish active and inactive channel users such that only active users will generate packets for transmission over the channel (with probability σ), the variable M in our model actually corresponds to the number of active users. In a real system, M will most probably vary during the day with alternate periods of heavy and light "load." Since such time periods are usually extremely large with respect to our time scale (a packet transmission time), M can be regarded as time-invariant during each period. A good rule of thumb in the system design is to optimize the channel performance under the assumption of a heavy load since the performance of a lightly loaded channel is relatively insensitive to the system design. This will be our philosophy in this chapter and the next. Most of our numerical examples are based upon the assumption of a heavily loaded channel. If we consider the average user think time to be 1-30 seconds in an interactive computer communications environment [JACK 69]. Our range of interest to be assumed for the number* of active channel users is between $M = 10$ to $M = 500$.

The mathematical model assumption that RD is geometrically distributed permits the use of a single variable for the state

---

*The user think time as defined in our model represents quantities such as the real thinking and typing time of an interactive terminal user or computer interburst time in the data stream model of Jackson and Stubbs [JACK 69]. The upper estimate $M = 500$ is obtained as follows. From our assumptions in Section 2.3.1 for a 50 KBPS channel, there are 44.4 time slots in one second. For an average user think time of 30 seconds, $\sigma = 1/(30 \times 44.4)$. From $M\sigma \leq 0.37$, we get $M \leq 0.37 \times 30 \times 44.4 \cong 500$. Note that our assumption of a 50 KBPS channel was quite arbitrary. If a higher channel data rate is considered (say 5 MBPS), we may want to assume different average think times to reflect a different type of users.

description. This assumption implies zero deterministic delay ($R = 0$). In a satellite channel this obviously represents an approximation. However, it is physically realizable in radio communications over short distances in which channel propagation delays are negligible compared to a packet transmission time. In this case, the duration of each channel time slot can be made longer to include R.

A satellite channel (such as considered in Section 2.3) has a round trip propagation delay of 0.27 seconds, which necessitates a state description consisting of the channel history for at least R consecutive time slots. The difficulty in mathematical analysis using such a state description was illustrated in the last chapter. Moreover, it was shown that the channel recovery time following an input pulse depends only upon the channel input rate and the channel backlog size. This observation provided the motivation for the current mathematical model. Below we show by simulations that the mathematical model also gives excellent prediction of the throughput-delay performance of a channel with nonzero R. The conclusion is that in most cases of interest, the slotted ALOHA channel performance is dependent primarily upon the expected value of the retransmission delay ($\overline{RD}$) and quite insensitive to the exact probability distributions considered.

In order to use the analytic results of the mathematical model to predict the throughput-delay performance of a slotted ALOHA channel with nonzero R, it is necessary to use a value of p in the mathematical model which gives the same $\overline{RD}$. For example, to approximate a slotted ALOHA channel with uniform retransmission

randomization, we must let

$$p = \frac{1}{R + (K + 1)/2} \tag{5.3}$$

such that $\overline{RD} = R + (K + 1)/2$ in both cases.

We define the length of time for which a packet is backlogged to be the backlog time of the packet and denote the <u>average backlog time</u> by D_b. To obtain the average packet delay as defined in Section 2.3 and illustrated in Fig. 2-2, we must add to D_b, $R + 1$ time slots, which represent the delay incurred by each successful transmission. Thus, we have

$$D = D_b + R + 1 \tag{5.4}$$

In the mathematical model $N^t = n$ implies that in the t^{th} time slot $(M - n)$ users are in the thinking state, each of which may generate and transmit a new packet with probability σ. Hence the channel input rate is $S^t = (M - n)\sigma$. However, when R is nonzero, the number of thinking users may be less than $(M - n)$, since some users may have had a successful transmission, but R time slots must pass by before they learn that "successful transmission occurred" (see Section 2.3). Suppose the channel throughput rate is S_{out}. By Little's result [LITT 61], there are on the average $S_{out} \cdot R$ such users (approximately equal to 4.5 for $R = 12$ and $S_{out} = \frac{1}{e}$) which is negligible when M is large (say a few hundreds). Moreover, the value of M can be adjusted to reflect this average value. For our purposes, this discrepancy will be ignored.

To show that the throughput-delay performance of a slotted ALOHA channel is dependent primarily upon $\overline{RD}$ and quite insensitive to its exact probability distribution, we performed a simulation experiment with the following probability distributions for RD :

(1) R = 12 and uniform randomization

(2) R = 0 and uniform randomization

(3) R = 12 and geometric randomization

(4) R = 0 and geometric randomization

The number of channel users M was assumed to be 200. The duration of each simulation run was 8000 slots. As in Chapter 3, only those simulation runs which satisfied our channel equilibrium criterion were considered. Two values of $\overline{RD}$ were used for each of the four cases: a large $\overline{RD}$ corresponding to K = 60 in case (1) and a small $\overline{RD}$ corresponding to K = 10 in case (1). Equivalent values of K and p giving the same $\overline{RD}$ were used for the other three cases. In cases (2) and (4), Eq. (5.4) was used to compute the average packet delay. The throughput-delay tradeoffs for all four cases at each of the two values of $\overline{RD}$ are shown in Fig. 5-1. Within the accuracy of the simulation experiment, all four cases give practically the same throughput-delay performance, lending validity to our claim that the channel throughput-delay performance is dependent upon the expected value rather than the exact probability distribution of RD . (Of course, in certain uninteresting situations such as K = 1 or 2 in case (1) or p very close to one in case (3), our claim is obviously invalid.)

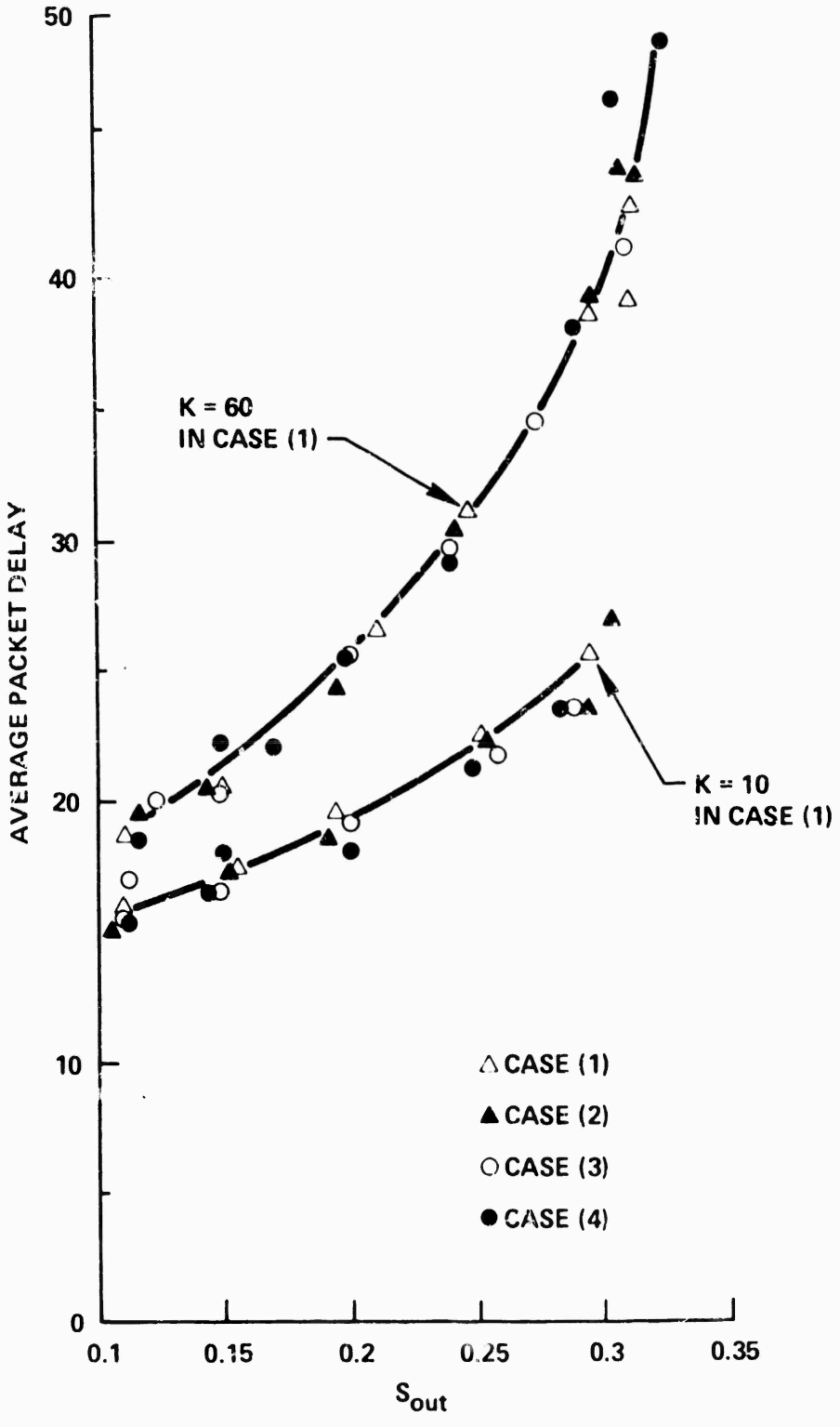


Figure 5-1. Comparison of Four RD Probability Distributions.

In this chapter and the next, the mathematical model as defined in the previous section will be studied. Use of Eqs. (5.3) and (5.4) enables the numerical results to be expressed in terms of K and compared with previous results on the slotted ALOHA channel performance for nonzero R and uniform retransmission randomization.

5.1.3 Channel Throughput

Conditioning on $N^t = n$, the expected channel throughput $S_{out}(n,\sigma)$ is the probability of exactly one packet transmission in the t^{th} time slot. Thus,

$$S_{out}(n,\sigma) = (1 - p)^n(M - n)\sigma(1 - \sigma)^{M-n-1} + np(1 - p)^{n-1}(1 - \sigma)^{M-n} \tag{5.5}$$

For the infinite population model, i.e., in the limit as $M \to \infty$ and $\sigma \to 0$ such that $M\sigma = S$ is finite and the channel input is Poisson distributed at the constant rate S, the above equation reduces to

$$S_{out}(n,S) = (1 - p)^n S e^{-S} + np(1 - p)^{n-1} e^{-S} \tag{5.6}$$

This expression is very accurate even for finite M if $\sigma << 1$ and if we replace $S = M\sigma$ by $S = (M - n)\sigma$. We assume that the condition $\sigma << 1$ is always satisfied in problems of interest to us.

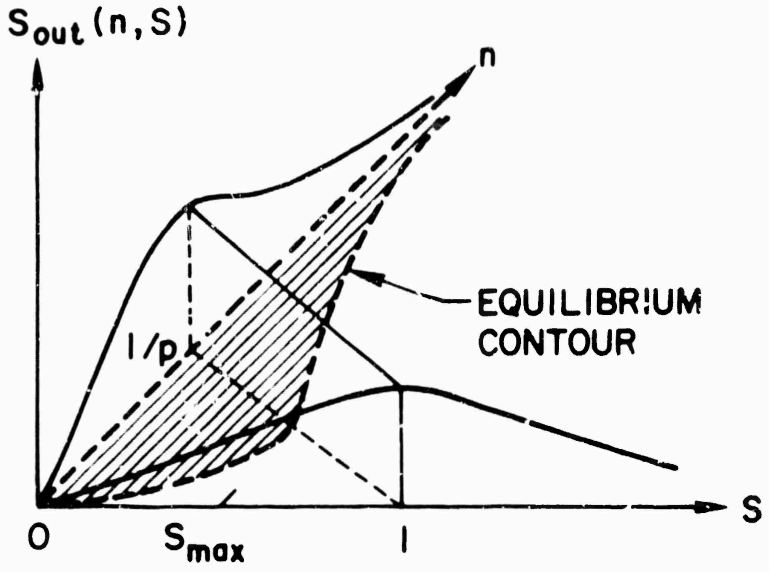


Figure 5-2. Channel Throughput Surface on the (n, S) Plane.

5.1.4 Equilibrium Contours

In Fig. 5-2, for a fixed K we show $S_{out}(n,S)$ as a 3-dimensional surface on the (n,S) plane given by Eq. (5.6). Note that there is an equilibrium contour in the (n,S) plane on which the channel input rate S is equal to the expected channel throughput $S_{out}(n,S)$. In the crosshatched region enclosed by the equilibrium contour, $S_{out}(n,S)$ exceeds S ; elsewhere, S is greater than $S_{out}(n,S)$ (the system capacity is exceeded!). In Fig. 5-3, a family of equilibrium contours for various K are displayed. We see that if we increase the average retransmission delay (by increasing K or equivalently decreasing p), these equilibrium contours move upwards. We show below that these equilibrium contours play a crucial role in determining the stability behavior of the channel.

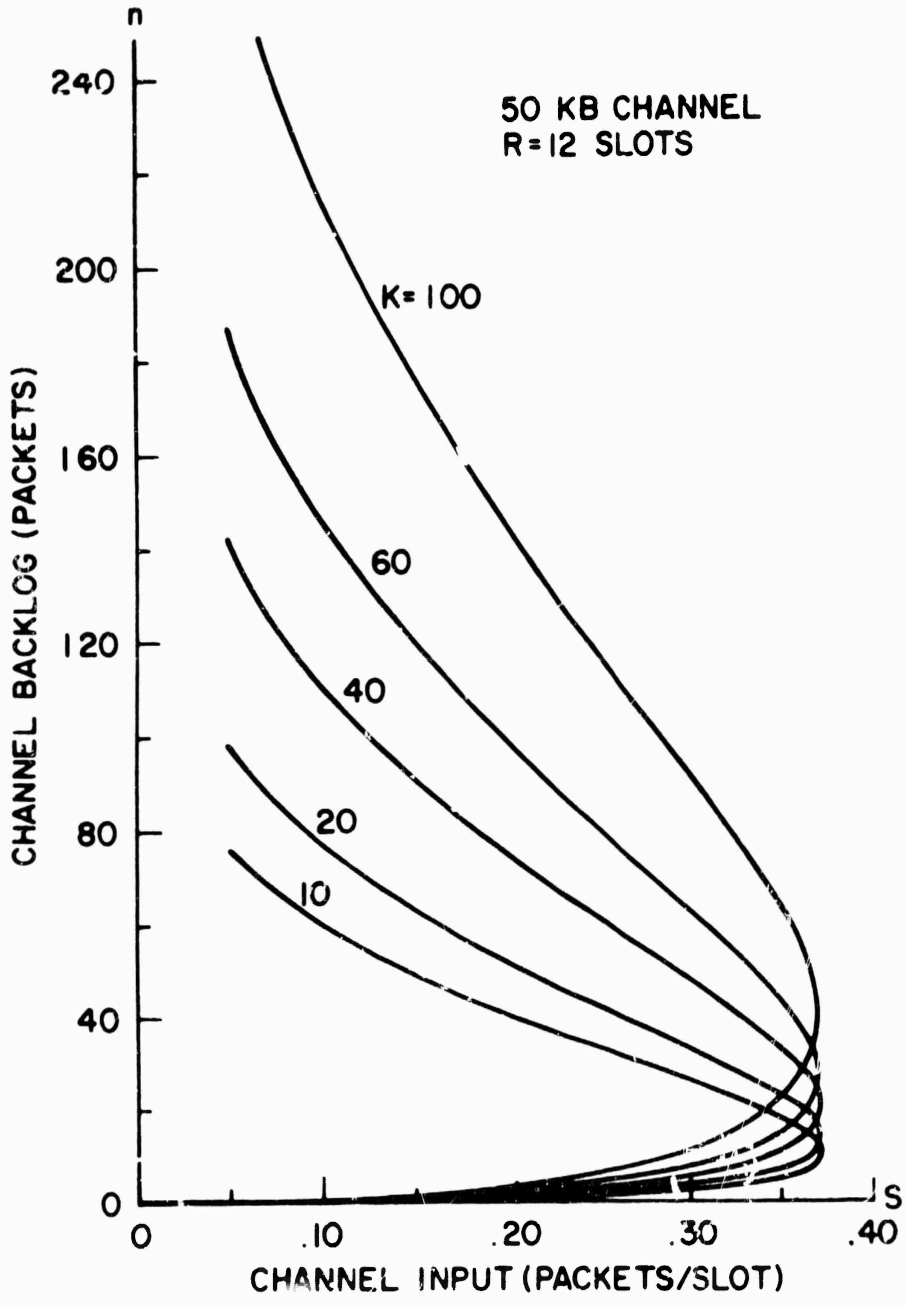


Figure 5-3. Equilibrium Contours on the (n, S) Plane.

Given M and σ and suppose a stationary probability distribution $\{P_n\}_{n=0}^{M}$ exists for N^t. Let $\bar{N} = \sum_{n=0}^{M} nP_n$. The stationary channel throughput rate S_{out} must be equal to the stationary channel input rate. That is,

$$S_{out} = \sum_{n=0}^{M} S_{out}(n,\sigma)\ P_n = \sum_{n=0}^{M} (M - n)\sigma\ P_n = (M - \bar{N})\sigma \tag{5.7}$$

For the equilibrium values of channel backlog size and throughput rate given by the condition $S_{out}(n,\sigma) = (M - n)\sigma$ to correctly predict the stationary average values $\bar{N}$ and S_{out}, a necessary condition is

$$S_{out}(\bar{N},\sigma) \cong \sum_{n=0}^{M} S_{out}(n,\sigma)\ P_n = (M - \bar{N})\sigma \tag{5.8}$$

For $p \ll 1$ and $\sigma \ll 1$, the above approximation is very accurate. For example, consider $K = 60$ and $M = 200$ in Fig. 5-8 below. The stationary channel throughput rate (computed by the value-determination operation in the next chapter) is found to be 0.344. The equilibrium value $S_0 = 0.346$.

Both the above equilibrium contours and the equilibrium contours shown in Figs. 3-3 and 3-4 in Chapter 3 are obtained under the condition that the channel input rate is equal to the channel throughput rate. Thus, a point specified by K and S in Fig. 5-3 must give rise to the same values of G and D in Figs. 3-3 and 3-4. Any discrepancy is due to the different approximations made in the two models (the first order approximation model and the linear feedback model).

The above claim can be verified by checking corresponding points on the contours. As an example, consider the point $K = 40$, $S = 0.275$ and $n = 54.5$ in Fig. 5-3. By Little's result [LITT 61], the average backlog time is

$$D = \frac{\bar{N}}{S_{out}}$$

Applying Eq. (5.4), we get $D = \frac{54.5}{0.275} + 13 = 211$ slots. Now if we check the corresponding point in Fig. 3-4 for $K = 40$ and $S = 0.275$, we find that $D = 212$ slots. In general, D values given by the linear feedback model are slightly less than those given by the first order approximation in Chapter 3. This is especially true when K is small such that the approximation in Eq. (5.8) becomes less accurate.

Channel state trajectories on the (n,S) plane

Given an equilibrium contour on the (n,S) phase plane, we consider here qualitatively the dynamic behavior of the channel subject to time-varying inputs. The following example serves to clarify similar fluid approximation results in Chapter 4.

Consider the case in which σ is constant while $M = M(t)$ is a function of time as shown in Fig. 5-4. We use the fluid approximation for the trajectory of the channel state vector (N^t, S^t) on the (n,S) plane as sketched in Fig. 5-5. Recall that $S^t = (M - N^t)\sigma$. The arrows indicate the "fluid" flow direction which depends on the relative magnitudes of $S_{out}(n,S)$ and S. Two possible cases are shown corresponding to different values of M_3 in Fig. 5-4. The

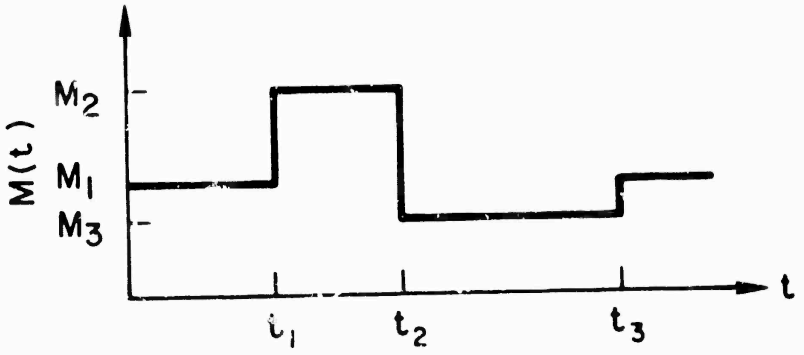


Figure 5-4. M(t)

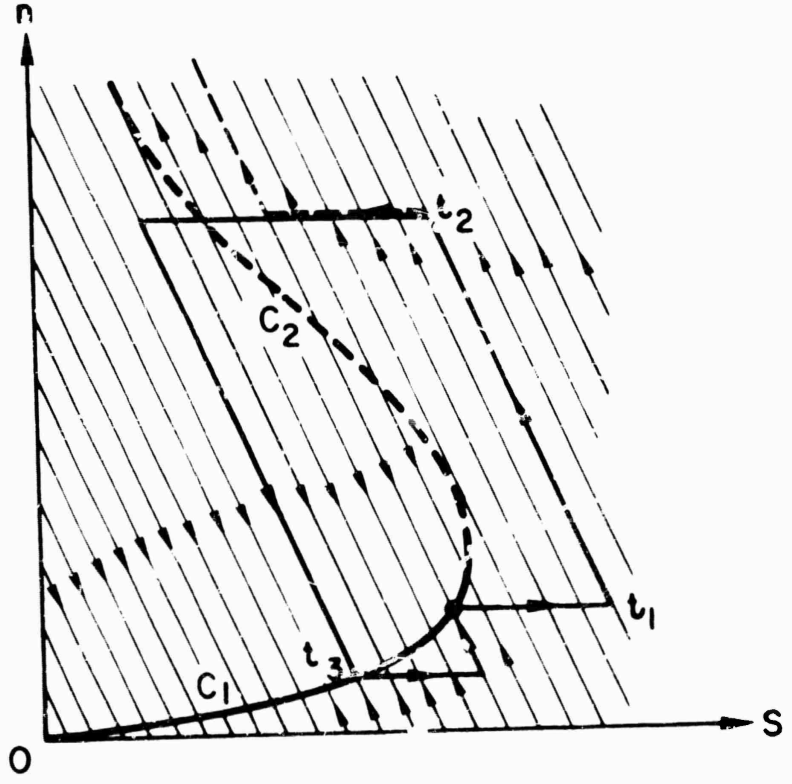


Figure 5-5. Fluid Approximation Trajectories.

solid line (case 1) represents a trajectory which returns to the original equilibrium point on contour C_1 despite the input pulse. The dashed line (case 2) represents a less fortunate situation in which the decrease in the channel input rate at time t_2 is not sufficient to bring the trajectory back into the "safe" region (in which $S < S_{out}(n,S)$). Eventually, the channel "collapses" as a result of an increasing backlog and a vanishing channel throughput rate. Compare these two cases with similar results in Figs. 4-1 and 4-2.

We have demonstrated channel saturation caused by a time-varying input. Next we study the conditions under which the slotted ALOHA channel with a stationary input (constant M and σ) can go into saturation as a result of statistical fluctuations.

5.2 Stability Considerations

We first define what we mean by stable and unstable channels and characterize their behavior. A stability measure is then given to quantify the relative instability of unstable channels.

5.2.1 Stable and Unstable Channels

Given M and σ, we define the channel load line in the (n,S) plane as the line $S = (M - n)\sigma$, which intercepts the n-axis at $n = M$ and has a slope equal to $-\frac{1}{\sigma}$.

The stability definition

The channel is said to be stable if its load line intersects (nontangentially) the equilibrium contour in exactly one place. Otherwise, the channel is said to be unstable.

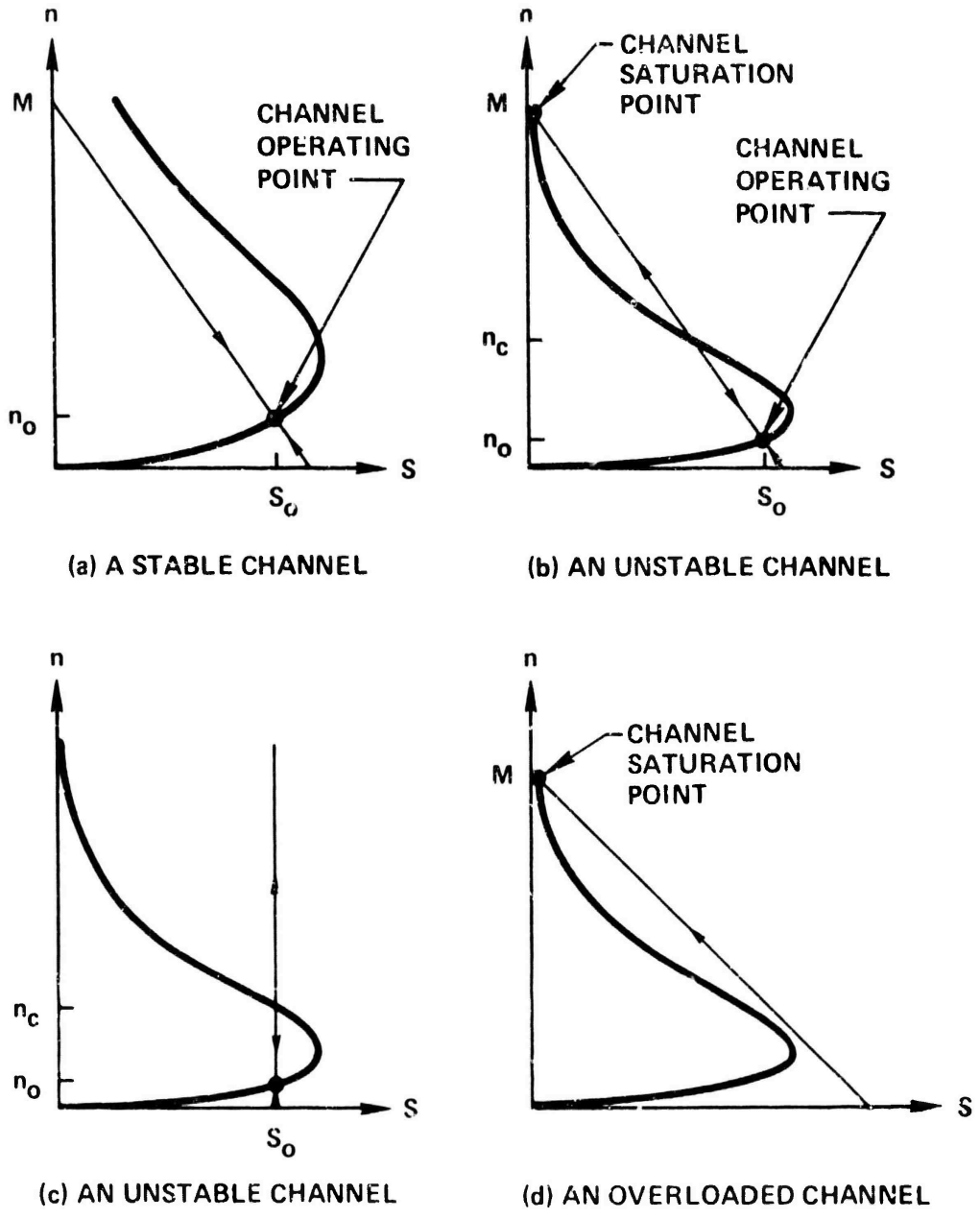


Figure 5-6. Stable and Unstable Channels.

Examples of stable and unstable channels are shown in Figs. 5-6. Arrows on the channel load lines indicate directions of fluid flow given by the fluid approximation. In other words, the arrow points in the direction of increasing backlog size if $S > S_{out}(n,S)$ and in the direction of decreasing backlog size if $S_{out}(n,S) > S$.

Each channel load line may have one or more equilibrium points. A point on the load line is said to be a stable equilibrium point if it acts as a "sink" with respect to fluid flow. It is a globally stable equilibrium point if it is the only stable equilibrium point on the channel load line. Otherwise, it is a locally stable equilibrium point. (Each stable equilibrium point is identified by a dot on channel load lines in Figs. 5-6 except in Fig. 5-6(c), where one of the stable equilibrium points is at $n = \infty$.) An equilibrium point is said to be an unstable equilibrium point if fluid flow emanates from it. Thus, the channel state N^t sitting on such a point will drift away from it given the slightest perturbation.

The stability definition given above is equivalent to defining a stable channel to be one whose channel load line has a globally stable equilibrium point.

In Fig. 5-6(a), we show the channel load line of a stable channel. Since N^t has a finite state space and is irreducible (assuming $p, \sigma > 0$), a stationary probability distribution always exists [PARZ 62]. Since (n_o, S_o) is the only equilibrium point on the load line, it gives the steady-state throughput-delay performance over an infinite time horizon under the approximation in Eq. (5.8). (n_o, S_o) will be referred to as the channel operating point. If M is finite, a stable channel can always be achieved

by using a sufficiently large K (see Fig. 5-3). Of course, a large K implies that the equilibrium backlog size n_o is large. As a result, the average packet delay may be too large to be acceptable.

In Fig. 5-6(b), we show the channel load line of an unstable channel. The point (n_o, S_o) is again the desired channel operating point since it yields the larger channel throughput and smaller average packet delay between the two locally stable equilibrium points on the load line. In fact, the other locally stable equilibrium point, having a huge backlog and virtually zero throughput, corresponds to channel saturation. It will be referred to as the channel saturation point. Since M is finite, and assuming $p, \sigma > 0$, a stationary probability distribution exists for N^t. However, N^t will "flip-flop" between the two locally stable equilibrium points in the following manner. Starting from an empty channel ($N^t = 0$ at time zero) quasi-stationary conditions will prevail at the operating point (n_o, S_o). The channel, however, cannot maintain equilibrium at this point indefinitely since N^t is a random process; that is, with probability one, the channel backlog N^t crosses the unstable equilibrium point n_c in a finite time and as soon as it does, the channel input rate S exceeds $S_{out}(n,S)$. Under this condition, N^t will drift toward the saturation point. (Although there is a positive probability that N^t may return below n_c, all our simulations showed that the channel state N^t accelerated up the channel load line producing an increasing backlog and a vanishing throughput rate.) Since the saturation point is a locally stable equilibrium point, quasi-stationary conditions will prevail there for some finite

(but probably very long) time period. In this state, the communication channel can be regarded as having failed. (In a practical system, external control should be applied at this point to restore proper channel operation.) The two locally stable equilibrium points on the load line of an unstable channel correspond to the channel being "up" or "down." An unstable channel may be acceptable if the average channel up time is large and external control is available to bring the channel up whenever it goes down.

In Figs. 5-7 and 5-8, we see how as the number of channel users M increases, an originally stable channel becomes unstable although the channel input rate S_o at the operating point remains constant (by reducing σ). For $S_o = 0.36$ and $K = 10$, we see that as M exceeds 80, the channel throughput decreases and the average packet delay increases very rapidly. (These results are obtained by solving for the stationary probability distribution of N^t using Algorithm 6.5 in the next chapter. No external control is assumed.) Using the stability definition and Fig. 5-3, the maximum value of M that is possible without rendering the load line unstable is $M_{max} = 79$, which exactly gives the knees of the curves in Fig. 5-7. In Fig. 5-8, by using a larger value of K $(= 60)$, a larger M_{max} is possible. Note, however, that the average packet delay ($\tilde{=}$ 56 slots) for $K = 60$ is much larger than the average packet delay ($\tilde{=}$ 36 slots) for $K = 10$. Given K and S_o, M_{max} can be obtained graphically from the equilibrium contours such as shown in Fig. 5-3. In Fig. 5-9, we show M_{max} as a function of K with S_o fixed at the maximum

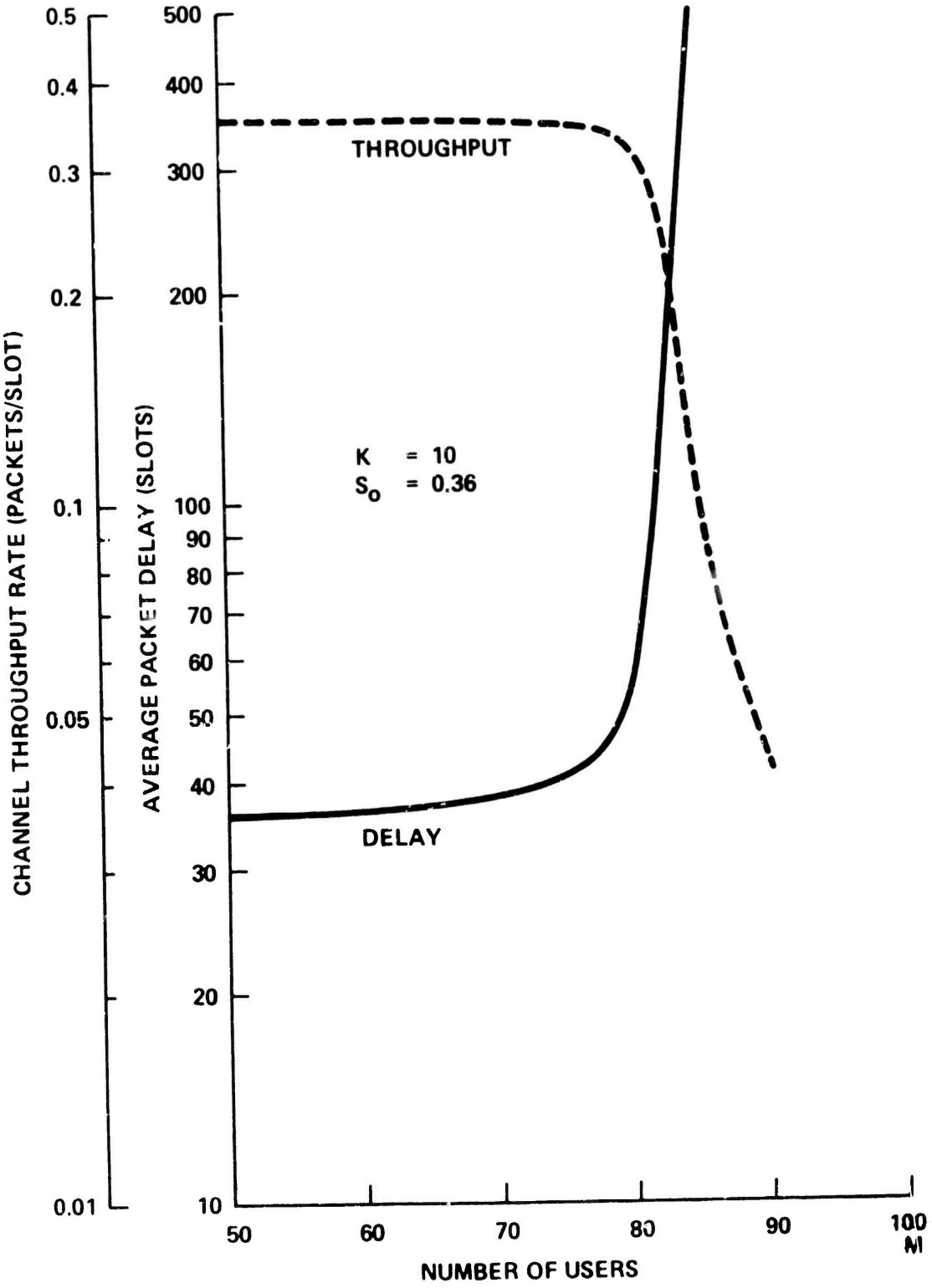


Figure 5-7. Channel Performance Versus M at K = 10 and S_o = 0.36

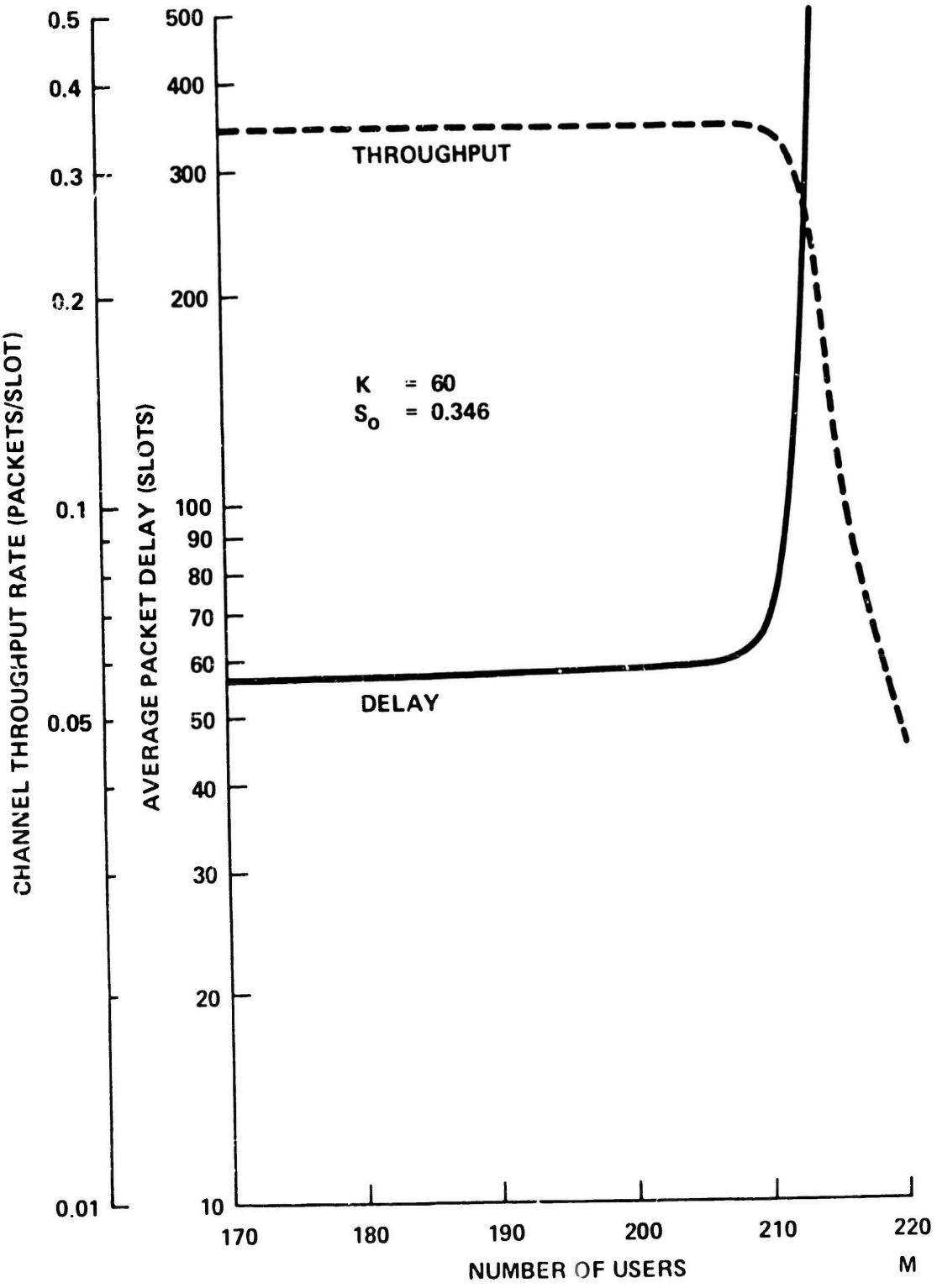


Figure 5-8. Channel Performance Versus M at K = 60 and S_o = 0.346

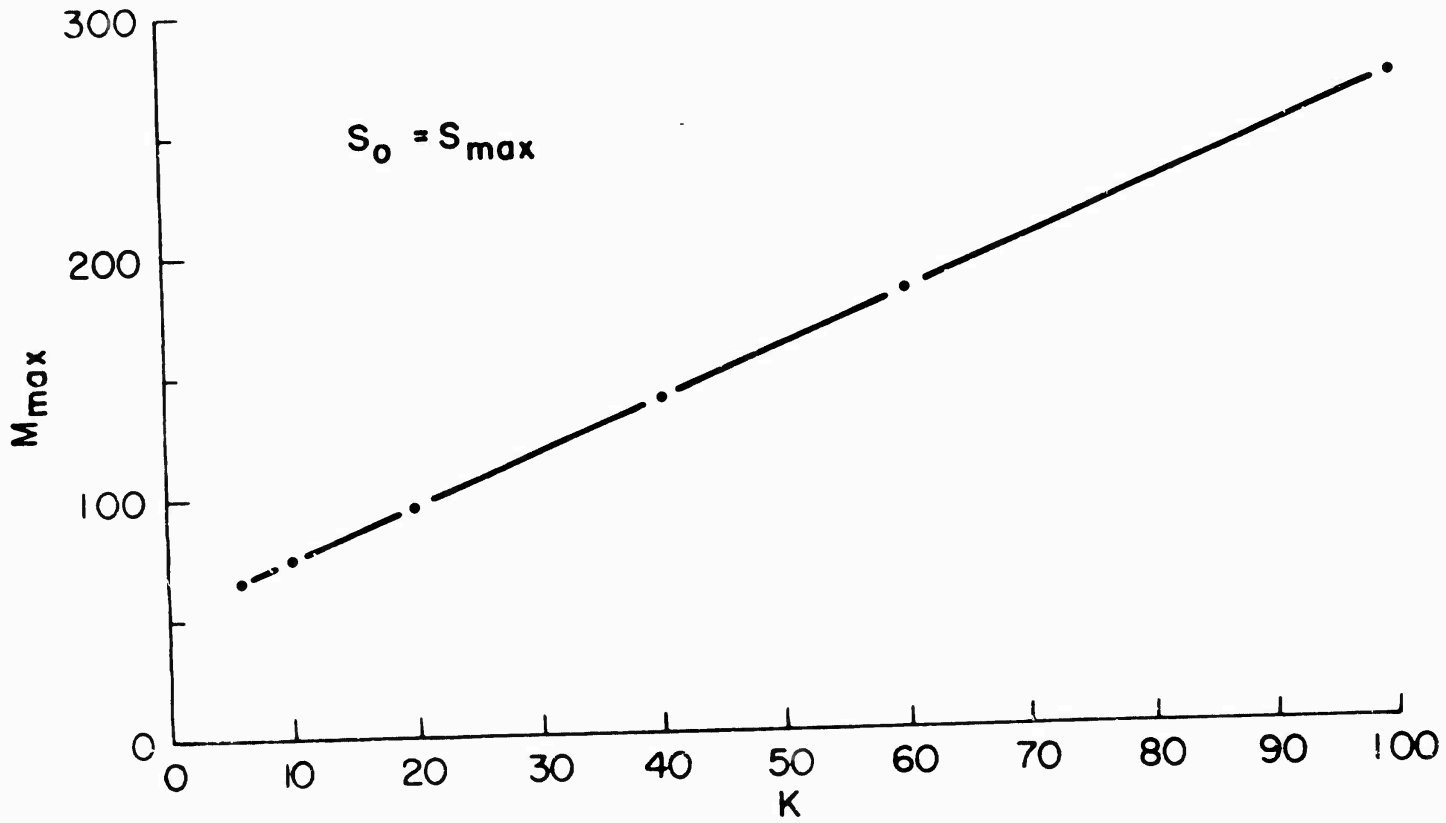


Figure 5-9. M_{max} Versus K.

value given K. Note the linear relationship between M_{max} and K for the values shown. In Fig. 5-10, we show that an originally unstable channel can be rendered stable by using a sufficiently large K.

The channel load line of an infinite population model is depicted in Fig. 5-6(c) as a vertical line. This is an unstable channel according to the stability definition. (Note that $n = \infty$ is a stable equilibrium point.) In fact, since N^t has an infinite state space and $S > S_{out}(n,S)$ for $n > n_c$, a stationary probability distribution does not exist for N^t. (See, for example, Cohen [COHE 69] pp. 543-546 for such proof.)

The channel load line shown in Fig. 5-6(d) is stable according to the stability definition. However, the globally stable equilibrium point in this case is the channel saturation point! Thus, this represents an "overloaded" channel as a result of bad system design. To correct this situation, the number of active users M supported by the channel should be reduced. Note that such an action is distinct from the dynamic control procedures in the next chapter, which are concerned with controlling temporary statistical fluctuations given that the channel is not overloaded in the above sense. From now on, a stable channel will always refer to the load line depicted in Fig. 5-6(a) instead of Fig. 5-6(d).

Let us summarize the major conclusions in the above discussions:

- The steady-state throughput-delay performance of a stable channel is given by its globally stable

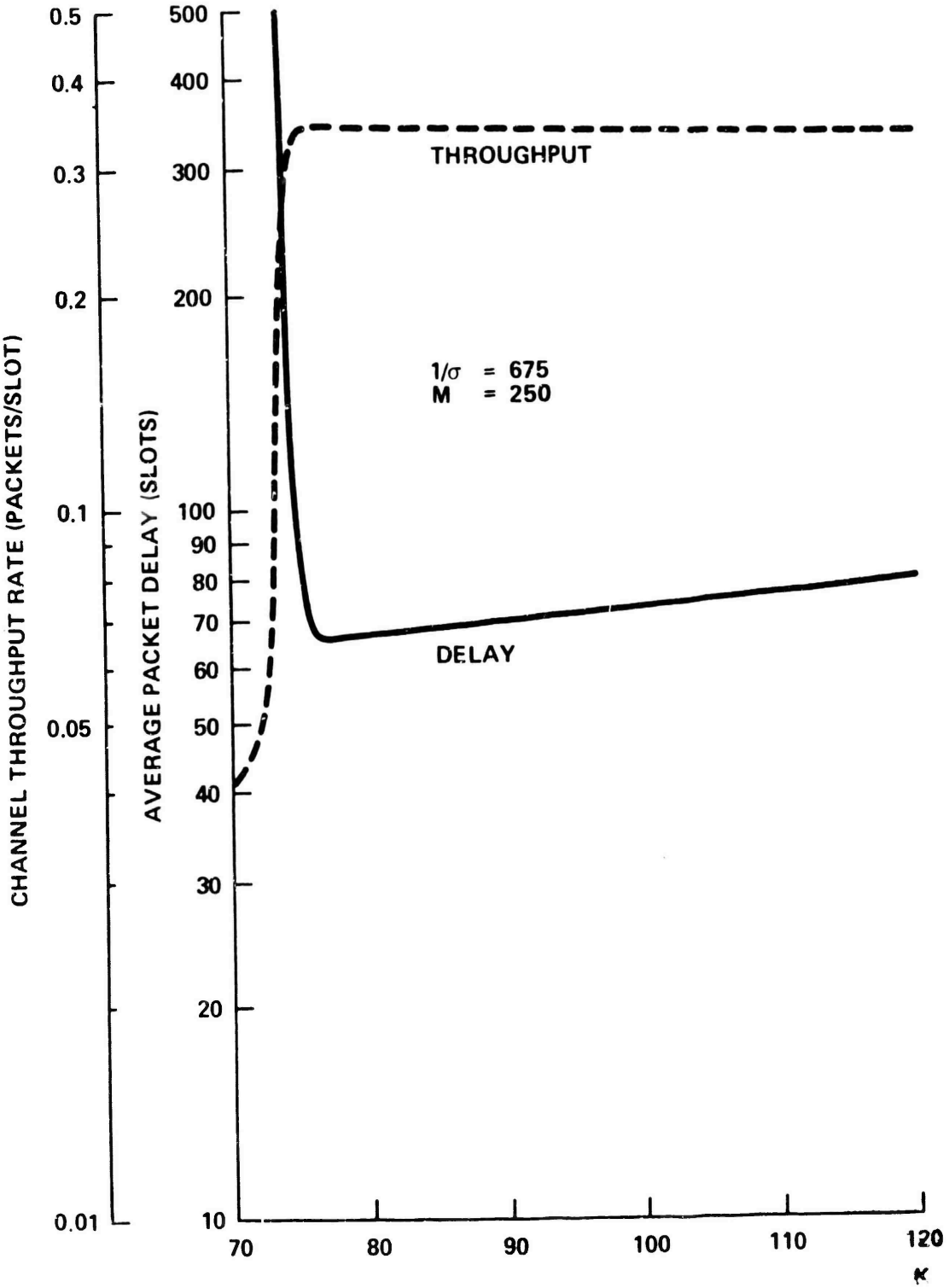


Figure 5-10. Channel Performance Versus K at M = 250 and $1/\sigma$ = 675.

equilibrium point and approximated by the equilibrium throughput-delay results in Chapter 3.

- In an unstable channel, the throughput-delay performance given by a locally stable equilibrium point can be achieved only for some finite time period.

5.2.2 A Stability Measure

From the above discussion and referring to Fig. 5-6(b), the load line of an unstable channel can be partitioned into two regions: the safe region consisting of the channel states $\{0, 1, 2, \ldots, n_c\}$ and the unsafe region consisting of the channel states $\{n_c + 1, \ldots, M\}$. A good stability measure (for these unstable channels!) is the average time to exit into the unsafe region starting from a safe channel state. To be exact, we define FET to be the average first exit time into the unsafe region starting from an initially empty channel ($N^t = 0$ at time zero). Thus, FET gives an approximate measure of the average up time of an unstable channel. Below we derive the probability distributions and expected values of such first exit times. The derivations are based upon well-known results on first entrance times in Markov chains with stationary transition probabilities [HOWA 71, PARZ 62].

Consider the mathematical model in Section 5.1 with constant M and σ, where M may be infinite. N^t is a Markov process (chain) with stationary transition probabilities $\{p_{ij}\}$ given by Eq. (5.1) or Eq. (5.2). Define the random variable T_{ij} to be the number of transitions which N^t goes through until it enters state j for the first time starting from state i. The probability

distribution of T_{ij} (called the first entrance probabilities from state i to state j) may be defined as

$$f_{ij}(m) = \text{Prob}[T_{ij} = m]$$

$$= \begin{cases} 0 & m = 0 \\ p_{ij} & m = 1 \\ \text{Prob}[N^{t+m} = j, N^{t+h} \neq j, h = 1, \ldots, m-1 \mid N^t = i] & m \geq 2 \end{cases} \tag{5.9}$$

The state space S for N^t consists of the set of non-negative integers $\{0, 1, 2, \ldots, n_c, n_c + 1, \ldots, M\}$ which is partitioned into the safe region $\{0, 1, 2, \ldots, n_c\}$ and the unsafe region $\{n_c + 1, \ldots, M\}$. Now consider the modified state space $S' = \{0, 1, 2, \ldots, n_c, n_u\}$ where n_u is an absorbing state such that N^t is now characterized by the transition probabilities

$$p_{ij}' = \begin{cases} p_{ij} & i, j = 0, 1, \ldots, n_c \\ \sum_{\ell=n_c+1}^{M} p_{i\ell} & i = 0, 1, \ldots, n_c \; ; \; j = n_u \\ 0 & i = n_u \; ; \; j = 0, 1, \ldots, n_c \\ 1 & i, j = n_u \end{cases} \tag{5.10}$$

Define the random variable T_i to be the number of transitions which N^t goes through before it enters the unsafe region for the

first time starting from state i in the safe region. T_i is called the first exit time from state i. The probability distribution of T_i is defined to be $\{f_i(m)\}_{m=1}^{\infty}$ which are called the first exit probabilities. It is trivial to show that starting from state i $(0 \le i \le n_c)$, the first entrance probabilities into the absorbing state n_u in the modified state space S' are the same as the first exit probabilities into the unsafe region of S. Using Eq. (5.9), such probabilities are given by the following recursive equation [HOWA 71],

$$f_{in_u}(m) = p'_{in_u}\,\delta(m-1) + \sum_{j=0}^{n_c} p'_{ij}\, f_{jn_u}(m-1) \qquad \begin{matrix} m \ge 1 \\ i \ne n_u \end{matrix}$$

where

$$\delta(m) = \begin{cases} 1 & m = 1 \\ 0 & \text{otherwise} \end{cases}$$

The above equation can be rewritten in terms of the first exit probabilities as

$$f_i(m) = \sum_{j=n_c+1}^{M} p_{ij}\,\delta(m-1) + \sum_{j=0}^{n_c} p_{ij}\, f_j(m-1) \qquad \begin{matrix} m \ge 1 \\ 0 \le i \le n_c \end{matrix} \tag{5.11}$$

where $f_i(m)$ can be solved recursively for $m \ge 1$ starting with $f_i(0) = 0$ for all i.

The probability distribution $\{f_i(m)\}_{m=1}^{\infty}$ for the random variable T_i typically has a very long tail and cannot be easily computed. We had defined earlier FET as a stability measure for an unstable channel. By our definition, FET is the same as the expected value of the random variable T_0. Let $\overline{T_i}$ be the expected value and $\overline{T_i^2}$ be the second moment of T_i. These moments can be obtained by solving a set of linear simultaneous equations. It can easily be shown [HOWA 71] that

$$T_i = \begin{cases} 1 & \text{with probability } p_{in_u} \\ 1 + T_j & \text{with probability } p_{ij} \end{cases}$$

from which we obtain [HOWA 71, PARZ 62]

$$\overline{T_i} = 1 + \sum_{j=0}^{n_c} p_{ij}\, \overline{T_j} \qquad i = 0, 1, \ldots, n_c \qquad (5.12)$$

$$\overline{T_i^2} = 2\,\overline{T_i} - 1 + \sum_{j=0}^{n_c} p_{ij}\, \overline{T_j^2} \qquad i = 0, 1, \ldots, n_c \qquad (5.13)$$

Eqs. (5.12) form a set of $n_c + 1$ linear simultaneous equations from which $\{\overline{T_i}\}_{i=0}^{n_c}$ can be solved and FET ($= \overline{T_0}$) determined. After $\{\overline{T_i}\}_{i=0}^{n_c}$ have been found, Eqs. (5.13) can then be solved in a similar manner for $\{\overline{T_i^2}\}_{i=0}^{n_c}$.

5.3 Numerical Results

With the stability measure defined above, we are now in a position to examine quantitatively the tradeoff among channel stability,

throughput and delay for unstable channels. Below we first give a computational procedure to solve for $\overline{T_i}$ and hence, FET. They are then computed for various values of K, S_o and M (corresponding to different load lines). The stability-throughput-delay tradeoff is then shown.

5.3.1 An Efficient Computational Algorithm

The solution of the set of simultaneous equations in either Eq. (5.12) or Eq. (5.13) involves inverting the $(n_c + 1)$ by $(n_c + 1)$ matrix in p_{ij} for $i, j = 0, 1, \ldots, n_c$. When n_c is large, this becomes a nontrivial task because of the large number of computational steps and large computer storage requirement for the $[p_{ij}]$ matrix. The fact that $p_{ij} = 0$ for $j \leq i - 2$ in Eqs. (5.1) and (5.2) enables us to use the following algorithm which is very efficient in terms of both the computer time and space requirements mentioned above when n_c is large.

Algorithm 5.1

This algorithm solves for the variables $\{t_i\}_{i=0}^{I}$ in the following set of $(I + 1)$ linear simultaneous equations,

$$t_0 = h_0 + \sum_{j=0}^{I} p_{0j}\, t_j$$

$$t_i = h_i + \sum_{j=i-1}^{I} p_{ij}\, t_j \qquad i = 1, 2, \ldots, I$$

(1) Define

$$e_I = 1$$

$$f_I = 0$$

$$e_{I-1} = \frac{1 - p_{II}}{p_{I,I-1}}$$

$$f_{I-1} = - \frac{h_I}{p_{I,I-1}}$$

(2) For $i = I - 1, I - 2, \ldots, 1$ solve recursively

$$e_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{I} p_{ij} \, e_j \right]$$

$$f_{i-1} = \frac{1}{p_{i,i-1}} \left[f_i - h_i - \sum_{j=i}^{I} p_{ij} \, f_j \right]$$

(3) Let

$$t_I = \frac{f_0 - h_0 - \sum_{j=0}^{I} p_{0j} \, f_j}{\sum_{j=0}^{I} p_{0j} \, e_j - e_0}$$

$$t_i = e_i \, t_I + f_i \qquad i = 0, 1, 2, \ldots, I-1$$

A derivation of the above algorithm is given in Appendix D. This algorithm is superior to conventional methods such as the Gauss elimination method [CRAI 64] for solving linear simultaneous equations in two respects. First, each p_{ij} is used exactly once and can be computed using Eq. (5.1) or Eq. (5.2) only when it is used in the

algorithm. This eliminates the need for storing the $[p_{ij}]$ matrix and practically eliminates any computer storage constraint on the dimensionality of the problem. Second, the number of arithmetic operations $(+ - \times \div)$ required by the above algorithm is in the order of $2I^2$ which is less than that of conventional methods such as Gauss elimination.

5.3.2 Average First Exit Times (FET)

In Fig. 5-11, we have shown FET as a function of K for the infinite population model and for fixed values of the channel throughput rate S_o (at the channel operating point). We see that FET can be improved by either decreasing the channel throughput rate S_o or by increasing K (which in turn increases the average packet delay) The infinite population model results give us the worst case estimates for channel stability as demonstrated in Fig. 5-12 in which we show FET as a function of M for $K = 10$ and four values of S_o. Note that FET increases as M decreases and there is a critical value of M below which the channel is always stable in the sense of Fig. 5-6(a). As M increases to infinity, FET reaches a limiting value corresponding to the infinite population model with a Poisson channel input. Fig. 5-13 is similar to Fig. 5-11 except now the number of users M is 150. Recall that if M is finite, the channel will become stable when K is sufficiently large.

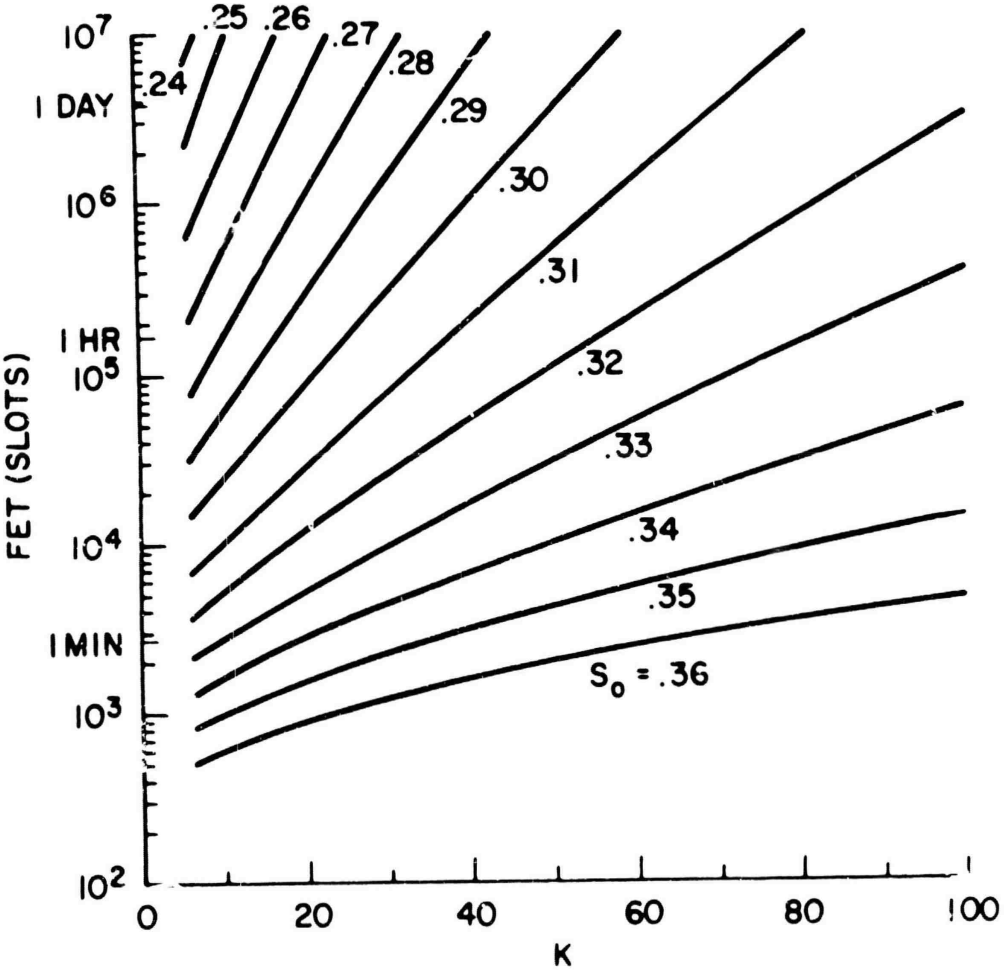


Figure 5-11. FET Values for the Infinite Population Model.

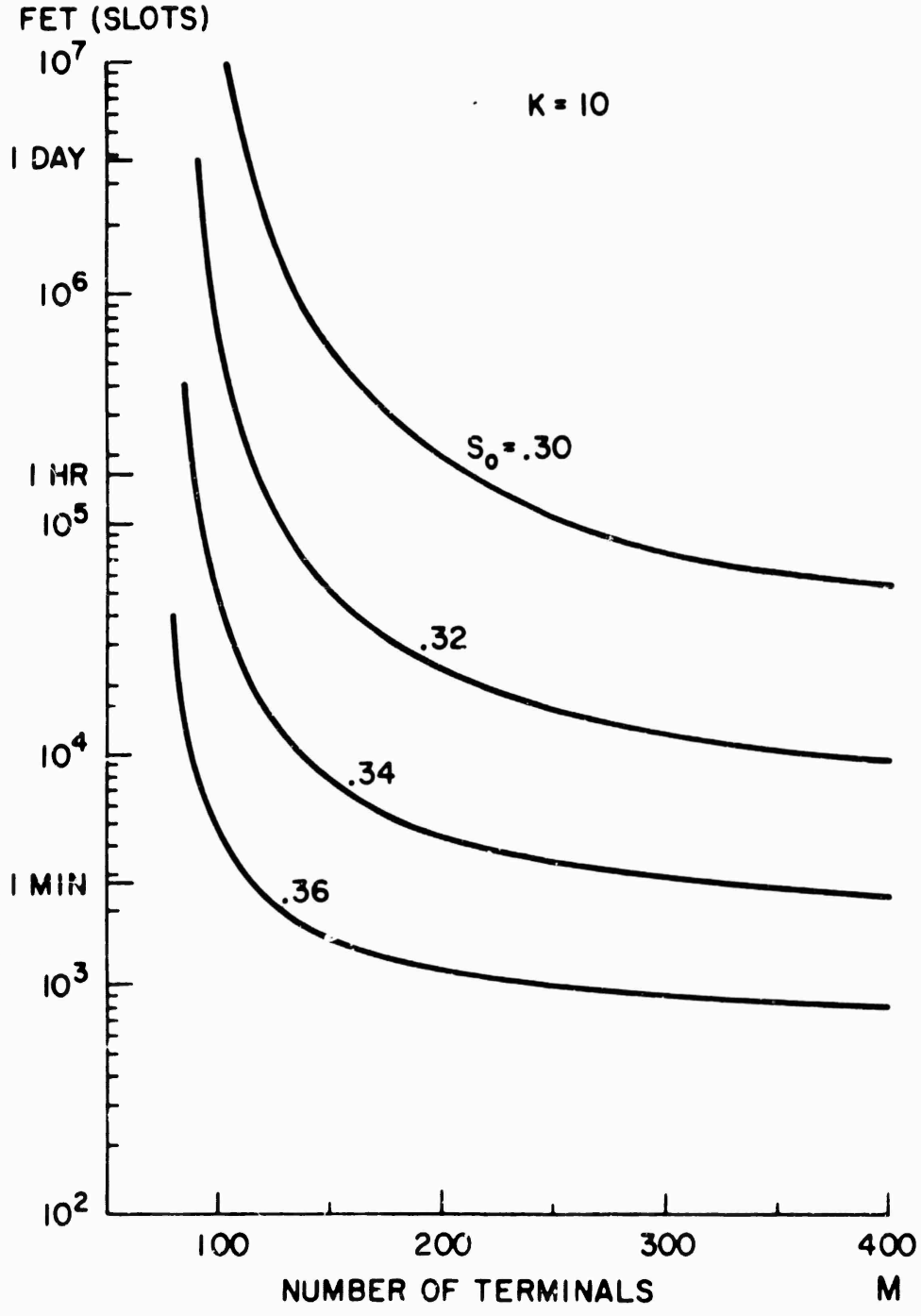


Figure 5-12. FET Versus M.

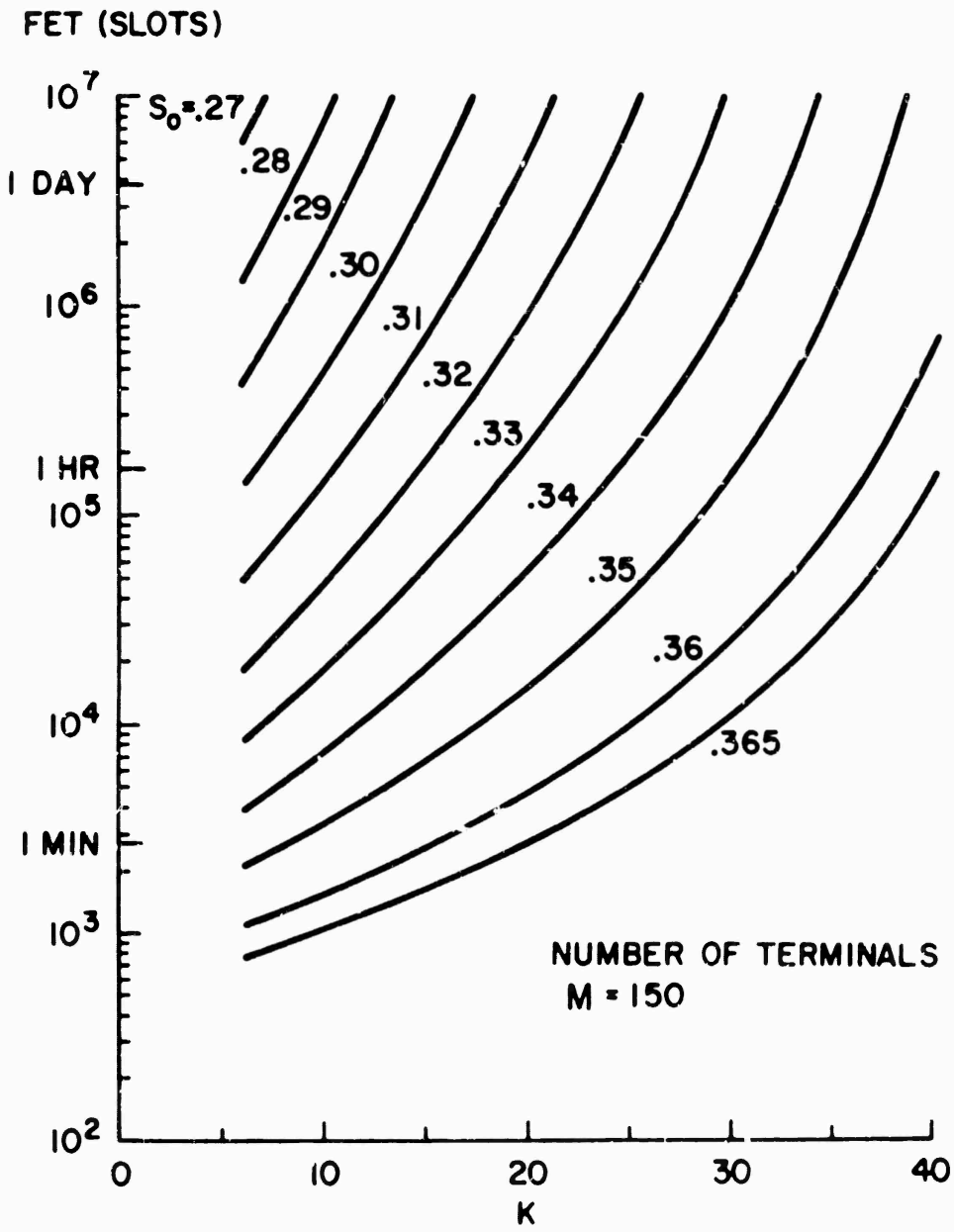


Figure 5-13. FET Values for a Finite User Population (M = 150)

As an example, we see that in Fig. 5-13 for $M = 150$, if the channel throughput rate S_o is kept at approximately 0.28 and $K = 10$ is used, the channel is estimated to fail once every two days on the average. If this is an acceptable level of channel reliability, then no other channel control procedure is necessary except to restart the channel whenever it goes into saturation. However, if absolute channel reliability is required at the same throughput-delay performance, dynamic channel control strategies should be adopted. Channel control schemes will be investigated in the next chapter.

5.3.3 The Stability-Throughput-Delay Tradeoff

In Fig. 5-14, we show as a lower bound the optimum performance envelope in Fig. 3-4 for the throughput-delay tradeoff of the infinite population model. This corresponds to the channel performance at the channel operating point indicated in Figs. 5-6. From these same figures, we see that the channel operating point (n_o, S_o) provides no information on the stability behavior of the channel. The equilibrium performance given by (n_o, S_o) is achievable in the long run if M is small enough such that the channel is stable; else, it is achievable only for some random time period estimated by our stability measure FET.

A design example

The designer of a slotted ALOHA channel is thus faced with the problem of deciding whether he wants a stable channel by using it for a small number of users and sacrifices channel utilization or uses the channel to support a large number of users if he is willing to accept

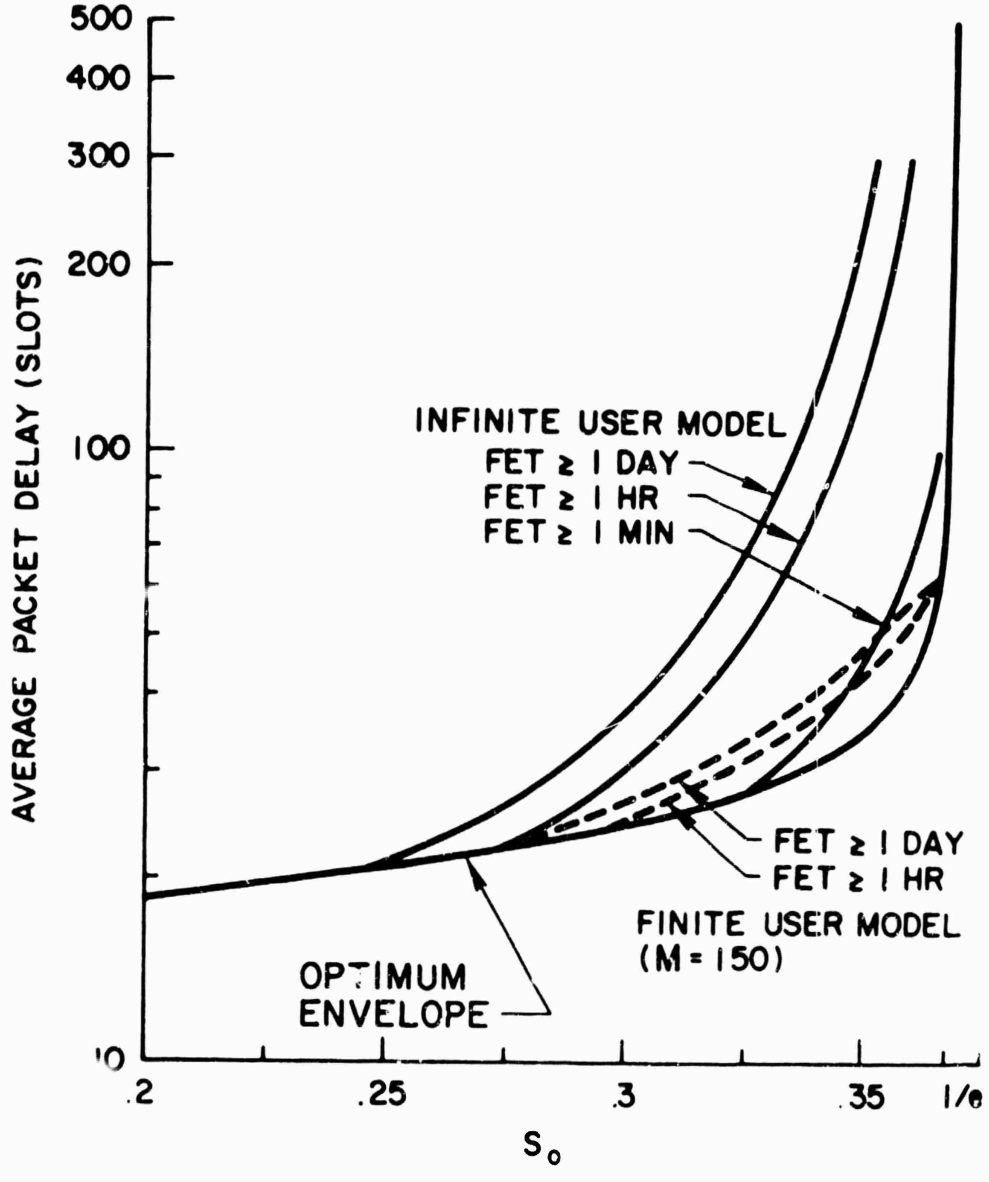


Figure 5-14. Stability-Throughput-Delay Tradeoff.

a certain level of channel reliability (some value of FET). For example, suppose K is chosen to be 10. (Note from Figs. 3-4 and 3-5 that K = 10 gives close to optimum equilibrium throughput-delay performance over a wide range of channel throughput rate.) Also, suppose that the channel users have an average think time of 20 seconds which, for our usual channel numerical constants, correspond to 888 time slots. Now if we draw channel load lines on Fig. 5-3 with a slope equal to - 888 , the channel is stable up to approximately 110 channel users. For M = 110 , the channel throughput rate S_o is about 0.125 packet/slot. From Fig. 3-4, the average packet delay is roughly 16.5 time slots (= 0.37 second). The same channel can be used to support 220 users at a channel throughput rate of S_o = 0.25 packet/slot. The average packet delay is 21 time slots (= 0.47 second). But now the channel is unstable! From Fig. 5-11, for K = 10 and S_o = 0.25 , the average up time (FET) of the channel is approximately two days for an infinite population model. Note that this value represents a lower bound for the FET of M = 220 . Thus, we see that if a channel failure rate of once every two days on the average is an acceptable level of reliability, the second channel design is much more attractive than the first since the number of channel users is more than doubled at a modest increase in delay.

In addition to the infinite population model optimum envelope, we also show in Fig. 5-14 two sets of equilibrium throughput-delay performance curves with guaranteed FET values. The first set consists of three solid curves corresponding to an infinite population model

with channel FET ≥ 1 day, 1 hour and 1 minute. Again, these results represent worst case estimates when M is finite. The second set consists of two dashed curves corresponding to $M = 150$ with channel FET ≥ 1 day and 1 hour. These results were obtained by looking up the values of K and S_o in Fig. 5-11 or Fig. 5-13 corresponding to a fixed FET. The average packet delay was then obtained from Fig. 3-4. This figure displays the fundamental tradeoff among channel stability, throughput and delay. In the next chapter, we devise strategies to dynamically control the channel to achieve truly stable throughput-delay performance close to the optimum performance envelope.

CHAPTER 6

DYNAMIC CHANNEL CONTROL

6.1 Introduction

Before we introduce channel control procedures, let us first examine the motivation for dynamic channel control.

In Chapter 1, we indicated that our interest in the multi-access broadcast channel stems from its capability to provide communication among a large population of users. In Chapter 3, equilibrium throughput-delay tradeoffs were given for the infinite population model (which approximates a large population of small users). The lower envelope of these tradeoffs characterizes the optimum channel performance. In Chapter 5, we showed that when the number of channel users M is sufficiently small, the channel is stable and the optimum channel performance envelope can actually be achieved over an infinite time horizon. However, for a large M, the channel is unstable. In this case, the optimum throughput-delay performance is achievable only for some finite time period before the channel goes into saturation.

In this chapter, we study dynamic channel control procedures which will enable an originally unstable slotted ALOHA channel not only to support a large number of users, but also to achieve a throughput-delay performance close to the optimum envelope with guaranteed channel stability.

The linear feedback model described in Section 5.1 is assumed throughout. In addition to this assumption, each channel user is

assumed to know the exact current channel state (channel backlog size). This assumption is necessary in the mathematical model, but will be relaxed when we consider heuristic (but practical) control procedures based upon the insights gained from the analysis.

Here we summarize the contents of this chapter. In Section 6.2, we give a brief introduction of Markov decision theory for a finite-state Markov process (chain) and outline Howard's policy-iteration method. Several control procedures are considered in Section 6.3. The first, known as the input control procedure (ICP), allows the channel to either accept or "reject" new packets from their sources. The second, known as the retransmission control procedure (RCP), allows the channel transmitters to impose either large or small retransmission delays on previously collided packets. The third, known as the input-retransmission control procedure (IRCP), is a combination of the first two as its name suggests. Two cost (performance) measures are defined, namely, the stationary channel throughput rate S_{out} and the average packet delay D. It will be shown in Section 6.4 that for each of the above control procedures, an optimal policy exists (and can be found by the policy-iteration method) which will maximize S_{out} and minimize D at the same time. An efficient computational algorithm is given in Section 6.5, which enables the use of the policy-iteration method for a large state space with relatively small computational and storage demands on the computer. Both numerical and simulation results are then given in Section 6.6 for the throughput-delay performance of the controlled random access

channel. In all cases considered, the optimal control policies were found to be of the control limit type. However, a rigorous proof of this result remains as an open problem.

In Section 6.7, we recognize the fact that the exact current channel state is not known to the individual channel users. A procedure is proposed which estimates the channel state and applies the above optimal control policies using this estimate. Another retransmission control procedure which circumvents the state estimation problem is also suggested. These control procedures are then tested through simulations and found to give not only a stable channel, but also achieve a throughput-delay performance close to the optimum performance envelope. Other channel control schemes proposed by Metcalfe [METC 73A] and Rettberg [RETT 73C] are then examined. Finally, we briefly discuss some channel design considerations.

6.2 Some Results from Markov Decision Theory

Most of the results in this section are taken from Howard [HOWA 60, HOWA 71] and Ross [ROSS 70]. Also, see Parzen [PARZ 62] for a general reference on Markov chains.

6.2.1 Markov Processes with Costs

We consider a finite Markov process (chain) N^t which is observed at time points $t = 0, 1, 2, \ldots$ to be in one of a finite number of possible states. The set of states S will be labelled by the nonnegative integers $\{0, 1, 2, \ldots, M\}$. The Markov process is assumed to have stationary state transition probabilities $\{p_{ij}\}$ (unless stated otherwise). The process incurs a cost c_{ij} when it

makes a transition from state i to state j. Thus, the Markov process starting at some initial state generates a sequence of costs as it makes transitions from state to state. Each c_{ij} is assumed to be bounded (i.e., $c_{ij} < \infty$) and independent of time (unless indicated otherwise).

We define C_i to be the expected immediate cost for state i and $v_i(\tau)$ to be the expected total cost that the process N^t incurs in the next $\tau + 1$ time units starting in state i. Hence,

$$C_i = \sum_{j=0}^{M} p_{ij} \, c_{ij} \tag{6.1}$$

$$v_i(\tau) = E\left[\sum_{t=0}^{\tau} C_{N^t} \;\middle|\; N^0 = i \right] \tag{6.2}$$

The expected total costs $v_i(\tau)$ are given by the following recurrence relation [HOWA 60]

$$v_i(\tau) = \sum_{j=0}^{M} p_{ij} \left[c_{ij} + v_j(\tau - 1) \right] \qquad \begin{matrix} i = 0, 1, 2, \ldots, M \\ \tau = 1, 2, 3, \ldots \end{matrix} \tag{6.3}$$

$$= C_i + \sum_{j=0}^{M} p_{ij} \, v_j(\tau - 1)$$

This set of equations can be solved recursively for the set of expected total costs $\{v_i(\tau)\}$ for any finite τ. However, when τ (called

the time horizon of the process $\{N^t\}$ is very large, a more suitable cost measure is the cost rate (i.e., expected cost per unit time) of the process. Thus, we define

$$g_i = \lim_{\tau \to \infty} \frac{1}{\tau + 1} E\left[\sum_{t=0}^{\tau} C_{N^t} \,\middle|\, N^0 = i \right] \tag{6.4}$$

where the limit always exists since the c_{ij} are bounded.

Assuming that N^t is an irreducible Markov chain and since S is finite, N^t possesses a unique stationary probability distribution $\{\pi_i\}_{i=0}^{M}$ such that [PARZ 62]

$$\pi_j = \sum_{i=0}^{M} \pi_i \, p_{ij} \qquad j = 0, 1, \ldots, M$$

$$\pi_i \geq 0 \qquad i = 0, 1, \ldots, M \tag{6.5}$$

and

$$\sum_{i=0}^{M} \pi_i = 1$$

From the ergodic theorems in the theory of Markov chains [CHUN 67], we then have the following important result

$$g = g_i = \sum_{j=0}^{M} \pi_j \, C_j \qquad \forall i = 0, 1, \ldots, M \tag{6.6}$$

where g is defined to be the cost rate or expected average cost of the process N^t and will be used extensively as the cost (performance) measure under various definitions of the state transition costs $\{c_{ij}\}$.

When τ is large, the expected total costs of the process, $v_i(\tau)$, are then given [HOWA 60] asymptotically by

$$v_i(\tau) = g\tau + v_i \qquad i = 0, 1, 2, \ldots, M \qquad (6.7)$$

where v_i is referred to as the asymptotic intercept of state i. For a large τ, $g\tau$ is the only significant variable. (However, it will be shown below that in a Markov decision process, relative values of the v_i will enable us to solve for an optimal control policy.)

6.2.2 Markov Decision Processes

We now introduce decision-making in the Markov process described above. Let A be a finite set of possible actions such that corresponding to each action $a \in A$, the set of state transition probabilities $\{p_{ij}(a)\}$ and costs $\{c_{ij}(a)\}$ (or equivalently the expected immediate costs $\{C_i(a)\}$) are uniquely specified. We define a policy f to be any rule for choosing actions and P to be the class of all policies. The action chosen by a policy at time t may, for instance, depend on the history of the process up to that point or it may be randomized in the sense that it chooses action a with some probability P_a, $a \in A$.

Suppose the action a^t is given by the policy f at time t, which in turn specifies the state transition probabilities and costs

at that time. Thus, f determines both the evolution in time of the Markov process N^t and the sequence of costs it incurs. For a policy f which generates the following sequence of actions in time $\{a^0, a^1, a^2, \ldots, a^t, \ldots\}$, we define the expected average cost per unit time for N^t which was initially in state i as

$$\phi_i(f) = \lim_{\tau \to \infty} \frac{1}{\tau + 1} E_f \left[\sum_{t=0}^{\tau} C_{N^t}(a^t) \mid N^0 = i \right] \tag{6.8}$$

where the limit always exists, since the costs are assumed to be bounded; the expectation is taken conditioning on the policy f. We say that the policy f^* is <u>average cost optimal</u> over all policies if $\phi_i(f^*) = \min_{f \in P} \phi_i(f)$ for all $i \in S$.

An important subclass of all policies is the <u>class of stationary policies</u> P_s. A <u>stationary policy</u> is defined to be one which is <u>nonrandomized</u> and the action it chooses at time t depends only on the state of the process at time t. Thus, a stationary policy f is a function $f(\cdot) : S \to A$. The Markov decision process employing a stationary policy f is in fact a Markov process with stationary transition probabilities and costs as described in the previous section. In this case, from Eq. (6.6)

$$\phi_i(f) = g(f) = \sum_{j=0}^{M} \pi_j(f) C_j(f) \qquad \forall i = 0, 1, \ldots, M \tag{6.9}$$

We give the following important result concerning stationary policies.

Theorem 6.1 Given a finite state space, if every stationary policy gives rise to an irreducible Markov chain, then there exists a stationary policy f^* which is optimal over the class of all policies. Thus,

$$g(f^*) = \phi_i(f^*) = \min_{f \in P} \phi_i(f) \qquad \forall i = 0, 1, \ldots, M$$

Proof See [ROSS 70].

The conditions in Theorem 6.1 will always be satisfied in our optimization problems below. Thus, by the above theorem, we can and shall limit our attention only to the class of stationary policies in our search for an optimal policy.

In the following section, we outline a procedure which solves for the cost rate g of a Markov decision process given a stationary policy f. An iteration method is then described, which leads to an optimal stationary policy within a finite number of iterations.

6.2.3 The Policy-Iteration Method [HOWA 60, HOWA 71]

Given a stationary policy f, the cost rate g of the resulting Markov process can be determined as follows. Substituting Eqs. (6.7) into Eqs. (6.3), we obtain

$$g + v_i = C_i + \sum_{j=0}^{M} p_{ij} v_j \qquad i = 0, 1, \ldots, M \qquad (6.10)$$

where the dependence of p_{ij} and C_i on the stationary policy f are suppressed. There are $(M + 2)$ unknown variables, namely, g

and $\{v_i\}$ in the $(M + 1)$ linear simultaneous equations. We see that Eqs. (6.10) are also satisfied if the v_i are replaced by $v_i + b$, where b is any arbitrary constant. Thus, although g can be determined uniquely, only relative values of the v_i can be obtained by solving Eqs. (6.10). Fortunately, g is the cost (performance) measure of the Markov process that we are interested in; a set of relative values of the v_i is sufficient for the purpose of the following iteration method in solving for an optimal policy.

The Policy-Iteration Method

The basic iteration cycle in the policy-iteration method is diagrammed below in Fig. 6-1.

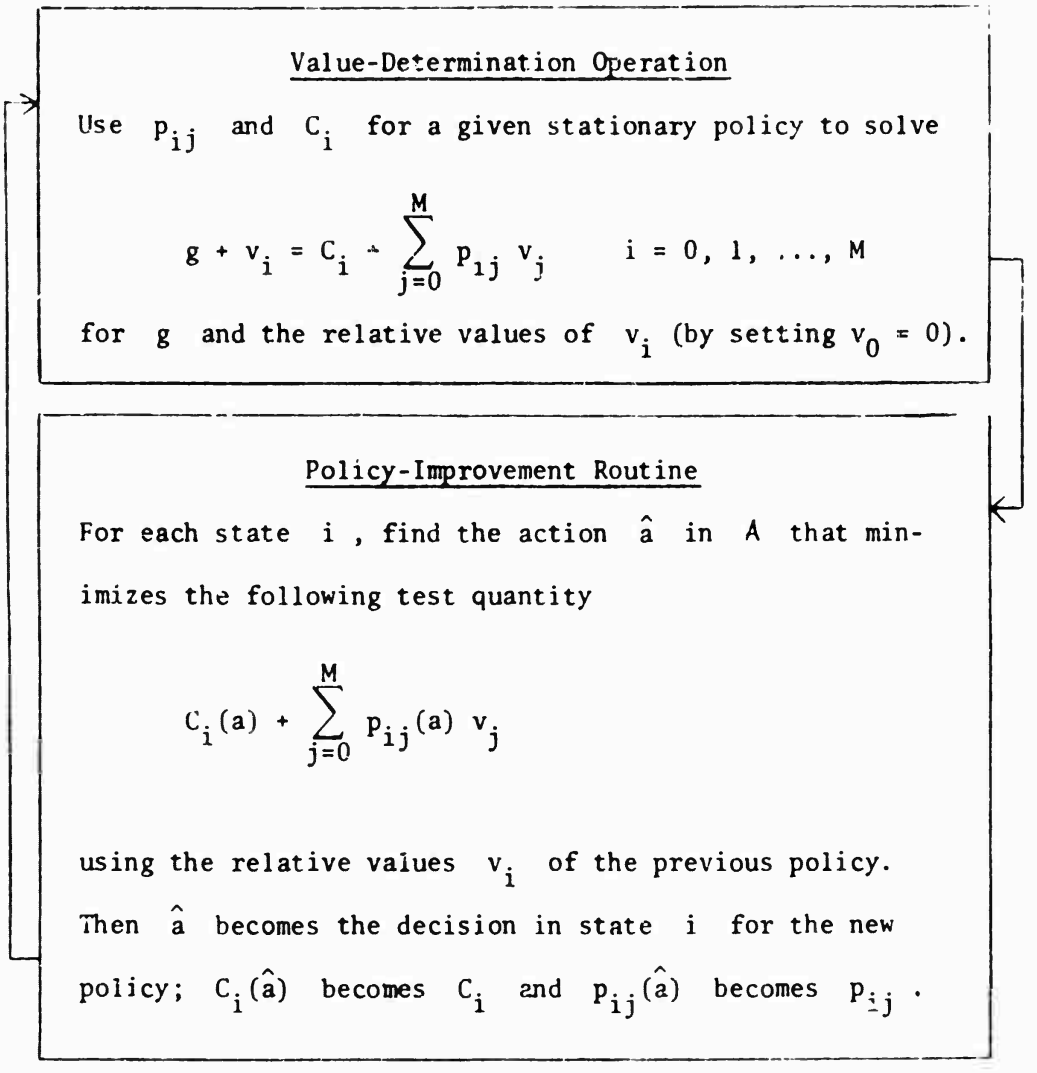


Figure 6-1 The policy-iteration cycle.

We may enter the iteration cycle in either box with an arbitrary initial policy or an arbitrary set of v_i. It is necessary to require that in the policy-improvement routine, if the decision $f(i)$ for state i given by the old policy yields as small a value for the test quantity as any of the other actions in A, the decision

is left unchanged. The stopping rule is as follows:

> The optimal policy has been reached (g is minimized) when the policies on two successive iterations are identical.

The following theorem on the policy-iteration method is due to Howard.

Theorem 6.2 (i) Suppose the policy-improvement routine has produced a policy f_2 that is different from the previous policy f_1, then

$$g(f_2) < g(f_1)$$

(ii) An optimal policy is obtained within a finite number of iterations.

Proof See [HOWA 60].

6.3 The Controlled Random Access Channel Model

Consider the stable and unstable channels in Figs. 5-6(a) and (b). The channel operating point (n_o, S_o) gives the throughput-delay performance of a stable channel. However, for an unstable channel, the throughput-delay performance given by the channel operating point* (n_o, S_o) is what we strive to achieve over an infinite time horizon through the use of dynamic channel control.

In this section, channel control procedures are proposed and formulated under the assumption that all channel users have perfect

---

*We assume that the channel operating point has been optimized over K (such that n_o is minimized) and that the optimal K has been adopted as the operating value of K.

knowledge of the current channel state (channel backlog size). We shall refer to this assumption as perfect channel state information.

6.3.1 The Markov Process

Consider the linear feedback model in Section 5.1, which represents a slotted ALOHA channel supporting input from M small independent users. The channel backlog size N^t at time t is taken to be the state variable with the state space $S = \{0, 1, 2, \ldots, M\}$. As before, we assume that each channel user in the thinking state generates and transmits a new packet independently with probability σ in a time slot; each channel user in the blocked state independently retransmits his backlogged packet with probability p in each time slot.* Thus, with constant M, σ and p, N^t is a finite-state Markov process with stationary state transition probabilities given by Eqs. (5.1) which we rewrite below.

$$p_{ij} = \begin{cases} 0 & j \le i - 2 \\ ip(1-p)^{i-1}(1-\sigma)^{M-i} & j = i - 1 \\ (1-p)^{i}(M-i)\sigma(1-\sigma)^{M-i-1} & \\ \qquad + \left[1 - ip(1-p)^{i-1}\right](1-\sigma)^{M-i} & j = i \\ \left[1 - (1-p)^{i}\right](M-i)\sigma(1-\sigma)^{M-i-1} & j = i + 1 \\ \binom{M-i}{j-i}\sigma^{j-i}(1-\sigma)^{M-j} & j \ge i + 2 \end{cases} \qquad 0 \le i, j \le M \tag{6.11}$$

*We again assume $p = \frac{1}{R + (K+1)/2}$ in our numerical computations as in Chapter 5. Our numerical results will be presented in terms of K so that they can be compared with previous results.

Cost rates and performance measures

The performance measures of interst to us are the stationary channel throughput rate S_{out} and average packet delay D. We show here how we can define the expected immediate costs C_i such that either S_{out} or D can be obtained from the resulting cost rate g of the Markov process.

Given that the Markov process N^t is in state i at time t, the expected channel throughput in the time slot is given by Eq. (5.5), which we rewrite below as

$$S_{out}(i) = ip(1 - p)^{i-1}(1 - \sigma)^{M-i} + (1 - p)^{i}(M - i)\sigma(1 - \sigma)^{M-i-1} \tag{6.12}$$

Now define the expected immediate (throughput) cost for state i as

$$C_i = - S_{out}(i) \tag{6.13}$$

and define the resulting cost rate of N^t as g_s. It can easily be shown from Eqs. (6.11) that N^t is aperiodic and irreducible for $p, \sigma > 0$. Thus, N^t has a stationary probability distribution $\{\pi_i\}$. Using Eq. (6.6), the stationary channel throughput rate is given by

$$S_{out} = \sum_{i=0}^{M} S_{out}(i)\ \pi_i = -g_s \tag{6.14}$$

Note that S_{out} must be equal to the stationary channel input rate

$$S = \sum_{i=0}^{M} (M - i)\, \sigma\, \pi_i \qquad (6.15)$$

To obtain the average packet delay D, we define the expected immediate (delay) cost for state i as

$$C_i = i$$

This accounts for the waiting cost of i packets incurred in the current time slot. In Markov decision theory terminology, this is sometimes referred to as the holding cost. Defining the resulting cost rate of the Markov process as g_d, we have from Eq. (6.6)

$$g_d = \sum_{i=0}^{M} i\, \pi_i$$

which is just the average channel backlog size $\bar{N}$ by definition. Applying Little's result [LITT 61], the average backlog time D_b of a packet is from Eq. (6.14)

$$D_b = \frac{\bar{N}}{S_{out}} = -\frac{g_d}{g_s}$$

and the average packet delay is from the above equation and Eq. (5.4)

$$D = -\frac{g_d}{g_s} + R + 1 \qquad (6.16)$$

where $R + 1$ represent the packet transmission time and propagation delay incurred by every packet in its successful transmission.

We note that the cost rates g_s and g_d can be obtained using the value-determination operation in the previous section given the appropriate definitions of the expected immediate costs C_i. The performance measures of interest S_{out} and D can then be computed from g_s and g_d using Eqs. (6.14) and (6.16).

6.3.2 Channel Control Procedures

By channel control procedure we mean the set of available actions in the action space A. Given the above Markov process formulation of the channel, we propose the following control procedures for which there exist policies which convert an unstable channel into a stable channel:

(1) The input control procedure (ICP)

(2) The retransmission control procedure (RCP)

(3) The input-retransmission control procedure (IRCP)

In Appendix F, we consider a general dynamic channel control procedure which includes ICP, RCP and IRCP as special cases.

The input control procedure (ICP)

This control procedure corresponds to the action space of the Markov decision process, $A = \{\text{accept, reject}\} \triangleq \{a,r\}$. Thus, in channel state i (i.e., given that $N^t = i$), the actions are:

accept (action = a) or reject (action = r) all new packets that arrive* in the current time slot.

The retransmission control procedure (RCP)

Under this control procedure, the action space $A = \{p_o, p_c\} \triangleq \{o, c\}$ where p_o and p_c are said to be the operating and control values of the retransmission probability p. (Through Eq. (5.3), p_o corresponds to K_o which gives the desired operating equilibrium contour and p_c corresponds to K_c which is large enough to render the channel stable.) Obviously, we must have $p_c < p_o$. Thus, in channel state i, the actions are: every backlogged packet is retransmitted in the current time slot with probability p_o (action = o) or with probability p_c (action = c).

In both control procedures, we see that channel stability is obtained through additional delays incurred by some or all packets in the system. However, they differ in their selection of such packets when the current channel state calls for "sacrifice" (i.e., choosing an action = r or c). In ICP, new packets are delayed ("rejected"); whereas in RCP, the backlogged packets are delayed for the social good.

The input-retransmission control procedure (IRCP)

This control procedure is a combination of ICP and RCP with the action space $A = \{(\text{accept}, p_o), (\text{accept}, p_c), (\text{reject}, p_o),$

---

*As discussed in Section 2.3.2, a new packet is said to arrive in the current time slot only after it has been generated by the channel user (or its external source), processed and ready for transmission over the channel in the current time slot. In the mathematical model, the rejected arrival is lost and the channel user generates a "new" packet in the next time slot with probability σ, etc. In a practical system, this new packet must actually be the previously rejected packet! We shall elaborate on this interpretation further below.

(reject, p_c)} $\triangleq$ {ao, ac, ro, rc} . Thus, for example, when the action rc is taken, both new and backlogged packets are delayed.

By Theorem 6.2, an optimal stationary policy can always be found. By virtue of Theorem 6.1, the optimal stationary policy given by the policy-iteration method is optimal over the class of all policies P for the given control procedure (action space A) . However, we do not claim that the control procedures we consider here give optimal policies over the class of all possible control procedures (action spaces). There are two reasons why we do not consider more multi-action control procedures other than IRCP.* (For example, RCP may be generalized so that $A = \{p_0, p_1, p_2, p_3\}$.) First, we realize that the channel state is in reality not exactly known but must be estimated. When A has many actions, the partitioning of the state space S induced by the control policy f may be too "fine" compared to estimation errors. Second, as we show below, the control procedures proposed above will give channel throughput-delay tradeoffs very close to the optimum envelope of the infinite population model (for which we ignored stability considerations). Hence, more elaborate control procedures will only give minute incremental improvement in channel performance.

A stationary policy can be defined by a function $f : S \to A$. For ICP, any stationary policy is uniquely specified by the sets S_a and S_r such that $S = S_a \cup S_r$, $S_a \cap S_r = \phi$ (the null set) and

$$f(i) = \begin{cases} a & i \in S_a \\ r & i \in S_r \end{cases} \qquad (6.7)$$

*A general dynamic control procedure is considered in Appendix F .

Similarly, a stationary policy of RCP is given by

$$f(i) = \begin{cases} o & i \in S_o \\ c & i \in S_c \end{cases} \qquad (6.18)$$

where $S_o \cup S_c = S$ and $S_o \cap S_c = \phi$

Within the class of stationary policies, a subclass of policies known as control limit policies can be described as follows for a two-action space A. Either the policy specifies the same action for all the states in S or there is a critical state $\hat{n}$ $(= 0, 1, 2, \ldots, M - 1)$ such that if the policy specifies one action for states 0 to $\hat{n}$, the other action is specified for states $\hat{n} + 1$ to M. $\hat{n}$ is said to be the control limit.

In Figs. 6-2 and 6-3, we show channel load lines corresponding to channels under ICP and RCP respectively. We find it easier to illustrate in both cases with control limit policies. In Fig. 6-2, $\hat{n}$ is the ICP control limit. When $N^t \leq \hat{n}$, the channel input rate $S^t = (M - N^t)\sigma$; when $N^t > \hat{n}$, $S^t = 0$. Similarly, suppose $\hat{n}$ is the RCP control limit in Fig. 6-3. When $N^t = n$, $K = K_o$, but as soon as N^t exceeds $\hat{n}$, $K = K_c$ is used. Note that both controlled channels are stable since the channel saturation point as shown in Fig. 5-6(b) no longer exists.

6.3.3 The Input Control Procedure (ICP)

Under this control procedure, recall that the action space $A = \{\text{accept, reject}\} = \{a, r\}$. We give below the state transition probabilities and costs of the Markov process $N^{\hat{n}}$ induced by each action in A.

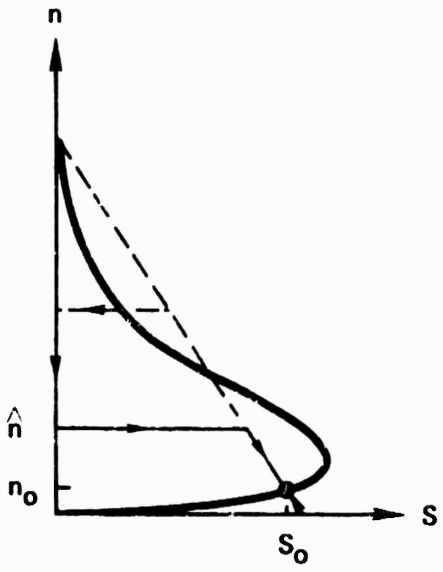


Figure 6-2. An ICP Control Limit Policy Example.

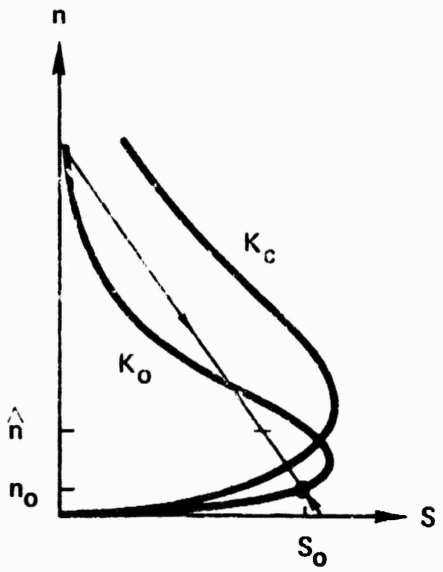


Figure 6-3. A RCP Control Limit Policy Example.

State transition probabilities

Suppose the channel is in state i $(= 0, 1, \ldots, M)$ and the stationary control policy $f(i) = a$ then $p_{ij}(a)$ is exactly as given in Eqs. (6.11), which we rewrite as

$$p_{ij}(a) = \begin{cases} 0 & j \le i - 2 \\ ip(1-p)^{i-1}(1-\sigma)^{M-i} & j = i - 1 \\ (1-p)^{j}(M-i)\sigma(1-\sigma)^{M-i-1} & \\ \quad + \left[1 - ip(1-p)^{i-1}\right](1-\sigma)^{M-i} & j = i \\ \left[1 - (1-p)^{i}\right](M-i)\sigma(1-\sigma)^{M-i-1} & j = i+1 \\ \binom{M-i}{j-i}\sigma^{j-i}(1-\sigma)^{M-j} & j \ge i + 2 \end{cases}$$

$$0 \le i, \; i < M \qquad (6.19)$$

Suppose $f(i) = r$, then $p_{ij}(r)$ is given as follows.

$$p_{ij}(r) = \begin{cases} ip(1-p)^{i-1} & j = i-1 \\ 1 - ip(1-p)^{i-1} & j = i \\ 0 & \text{otherwise} \end{cases} \tag{6.20}$$

Except in the uninteresting cases when σ, $p = 0$ or $f(i) = r$ for all $i \in S$, the Markov process N^t under this control procedure is aperiodic and irreducible satisfying the conditions of Theorem 6.1.

Rejection costs

As in the Markov process formulation of an uncontrolled channel described in Section 6.2.1, expected immediate costs are incurred in every time slot. Depending on the performance measure (D or S_{out}), there is a holding cost which pertains to packet delays and there is a negative cost which is the expected channel throughput in that time slot. With ICF, we also introduce the rejection cost d_r which is the expected cost in units of delay per packet arrival rejected.

For an interpretation of this cost in terms of its effect on packet delays, we consider as an example the possible terminal access communications environment depicted in Fig. 6-4. A person sitting at a terminal generates a new packet with an average think time of $\frac{1}{\sigma}$ whenever his previous packet has been successfully transmitted. If, at the time of a packet arrival, the channel is in the reject state, this packet is lost in the sense that it is not transmitted

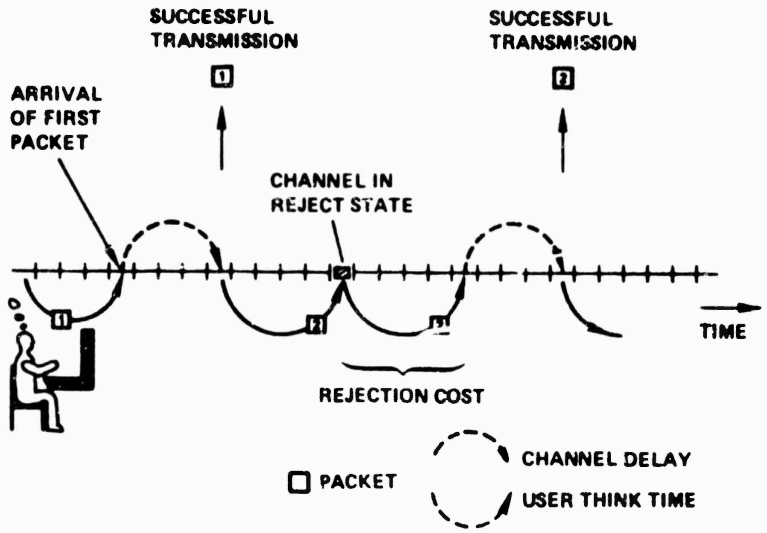


Figure 6-4. An Interpretation of the Rejection Cost.

over the broadcast channel at this time. In a practical situation, the user may be informed of the event and must enter some command character to "restart" the packet. Hence, the cost in terms of delay is probably in the order of an average think time ($= \frac{1}{\sigma}$) . Let

$$d_r = \frac{\alpha}{\sigma} \tag{6.21}$$

We shall assume $\alpha = 1$ throughout this chapter. This assumption is actually necessitated by our Markov process model in Section 6.3.1, where each thinking user is assumed to transmit a new packet (which may be a previously rejected new packet) with probability σ in a time slot.

It is easy to think of situations in which α is not one. For example, we may want to insert additional delays to rejected

packets or to account for some terminal processing time by using $\alpha > 1$. On the other hand, the human user may be very impatient and restarts his rejected packets very quickly such that $\alpha < 1$. (In this case, the terminal can always insert additional delays to make $\alpha = 1$.) In any case, if $\alpha \neq 1$, our channel state description will become more complex since we must distinguish blocked users who transmit in a time slot with probability p, thinking users with rejected packets who transmit with probability $\frac{\sigma}{\alpha}$ and the other thinking users who transmit with probability σ. Assuming $\alpha = 1$ in ICP (and also IRCP) simplifies the state description and consequently the amount of computation required in the policy-iteration method.

Average packet delay and channel throughput rate

Consider a stationary control policy $f : S \to A$ uniquely specified by the sets S_a and S_r. The expected immediate (delay) costs for state i are (assuming $\alpha = 1$)

$$C_i(a) = i \tag{6.22}$$

$$\begin{aligned} C_i(r) &= i + (M - i)\,\sigma\, d_r \\ &= i + (M - i) \\ &= M \end{aligned} \tag{6.23}$$

From Eq. (6.9), the cost rate of the process N^t under policy f is given by

$$g_d(f) = \sum_{i=0}^{M} \pi_i(f)\, C_i(f)$$

$$= \sum_{i=0}^{M} i\, \pi_i(f) + d_r \sum_{i \in S_r} (M - i)\sigma\, \pi_i(f) \qquad (6.24)$$

where $\{\pi_i(f)\}$ are the stationary probabilities of the process N^t whose state transition probabilities $\{p_{ij}(f)\}$ are given by Eqs. (6.19) and (6.20). Define

$$\lambda_r = \sum_{i \in S_r} (M - i)\sigma\, \pi_i(f) \qquad (6.25)$$

to be the rate of packet rejection for all the channel users. Thus, Eq. (6.24) can be rewritten as

$$g_d(f) = \sum_{i=0}^{M} i\, \pi_i(f) + \lambda_r\, d_r$$

$$= \bar{N} + \overline{N_r} \qquad (6.26)$$

where by Little's result [LITT 61], $\bar{N}$ is the average channel backlog size and $\overline{N_r}$ is the average number of rejected packets in the system. Considering Fig. 6-5 and applying Little's result once more, the average packet delay (including rejection delays) is given by

$$D = \frac{g_d(f)}{S_{out}} + R + 1 \qquad (6.27)$$

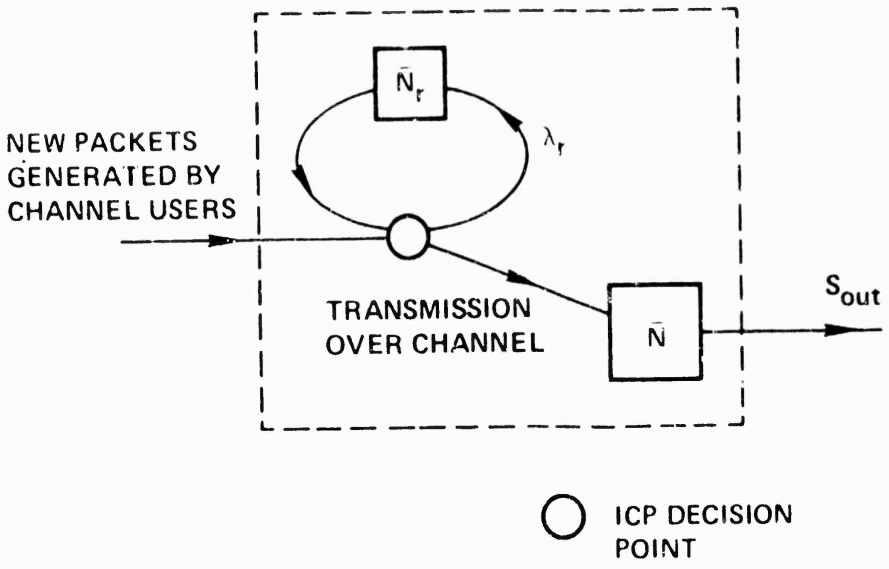


Figure 6-5. Average Number of Packets in the System Under ICP.

where as before $R + 1$ account for the packet transmission time and propagation delay for the successful transmission and S_{out} is the stationary channel throughput rate to be obtained in the following manner.

Given policy f as above, we define the following expected immediate (throughput) costs

$$C_i(a) = -S_{out}(i, a)$$

$$= -[ip(1-p)^{i-1}(1-\sigma)^{M-i} + (1-p)^{i}(M-i)\sigma(1-\sigma)^{M-i-1}] \qquad (6.28)$$

$$C_i(r) = -S_{out}(i, r)$$

$$= -ip(1-p)^{i-1} \qquad (6.29)$$

Using the above definitions, the cost rate of the Markov process N^t is by Eq. (6.9)

$$g_s(f) = - \sum_{i=0}^{M} \pi_i(f)\, S_{out}(i, f)$$

Thus, the stationary channel throughput rate is

$$S_{out} = - g_s(f) \tag{6.30}$$

The average packet delay is from Eq. (6.27)

$$D = - \frac{g_d(f)}{g_s(f)} + R + 1 \tag{6.31}$$

Given f, $g_d(f)$ and $g_s(f)$ can be calculated using the value-determination operation in the policy-iteration method assuming delay and throughput costs respectively.

6.3.4 The Retransmission Control Procedure (RCP)

Under this control procedure, the action space $A = \{p_o, p_c\} = \{o, c\}$. We give below the state transition probabilities and costs of the Markov process N^t induced by each action in A.

State transition probabilities

Suppose the channel is in state i (= 0, 1, ..., M) and action p_o is selected, then $p_{ij}(o)$ is given by

$$p_{ij}(o) = \begin{cases} 0 & j \le i - 2 \\ i\, p_o(1 - p_o)^{i-1}(1 - \sigma)^{M-i} & j = i - 1 \\ (1-p_o)^i(M-i)\sigma(1-\sigma)^{M-i-1} + [1-ip_o(1-p_o)^{i-1}](1-\sigma)^{M-i} & \\ & j = i \\ [1 - (1 - p_o)^i](M - i)\sigma(1 - \sigma)^{M-i-1} & j = i + 1 \\ \binom{M - i}{j - i} \sigma^{j-i} (1 - \sigma)^{M-j} & j \ge i + 2 \end{cases} \tag{6.32}$$

If action p_c is selected, then $p_{ij}(c)$ is given by

$$p_{ij}(c) = \begin{cases} 0 & j \le i - 2 \\ i\, p_c(1 - p_c)^{i-1}(1 - \sigma)^{M-i} & j = i - 1 \\ (1-p_c)^i(M-i)\sigma(1-\sigma)^{M-i-1} + [1-ip_c(1-p_c)^{i-1}](1-\sigma)^{M-i} & \\ & j = i \\ [1 - (1 - p_c)^i](M - i)\sigma(1 - \sigma)^{M-i-1} & j = i + 1 \\ \binom{M - i}{j - i} \sigma^{j-i} (1 - \sigma)^{M-j} & j \ge i + 2 \end{cases} \tag{6.33}$$

Except in the uninteresting cases when σ , p_o or $p_c = 0$, the Markov process N^t under this control procedure is aperiodic and irreducible satisfying the conditions of Theorem 6.1.

Average packet delay and channel throughput rate

Consider a stationary control policy $f : S \to A$ uniquely specified by the sets S_o and S_c. The expected immediate (delay) cost in state i is just the holding cost for both actions,

$$C_i(o) = C_i(c) = i \tag{6.34}$$

As before, the resulting cost rate of N^t is given by Eq. (6.9)

$$g_d(f) = \sum_{i=0}^{M} \pi_i(f) C_i(f) = \sum_{i=0}^{M} i\, \pi_i(f) = \bar{N} \tag{6.35}$$

Thus, the average packet delay is given by Eq. (6.27)

$$D = \frac{g_d(f)}{S_{out}} + R + 1 \tag{6.27}$$

where S_{out} is the stationary channel throughput rate.

The expected immediate (throughput) costs are given by

$$C_i(o) = - S_{out}(i, o)$$

$$= - \left[i\, p_o (1-p_o)^{i-1} (1-\sigma)^{M-i} + (1-p_o)^{i} (M-i) \sigma (1-\sigma)^{M-i-1} \right]$$

(6.36)

$$C_i(c) = - S_{out}(i, c)$$

$$= -[i\, p_c(1-p_c)^{i-1}(1-\sigma)^{M-i} + (1-p_c)^i (M-i)\sigma(1-\sigma)^{M-i-1}]$$

(6.37)

Using the above definitions, the cost rate of the Markov process N^t is given by

$$g_s(f) = - \sum_{i=0}^{M} \pi_i(f)\, S_{out}(i, f)$$

Thus, the stationary channel throughput rate is again

$$S_{out} = - g_s(f) \tag{6.30}$$

and the average packet delay is

$$D = - \frac{g_d(f)}{g_s(f)} + R + 1 \tag{6.31}$$

6.3.5 The Input-Retransmission Control Procedure (IRCP)

This control procedure is a combination of ICP and RCP. The action space $A = \{(\text{accept}, p_o), (\text{accept}, p_c), (\text{reject}, p_o), (\text{reject}, p_c)\} = \{ao, ac, ro, rc\}$. We give below the state transition probabilities and costs of the Markov process N^t induced by each action in A.

State transition probabilities

For $i = 0, 1, 2, \ldots, M$

$$p_{ij}(ao) = \begin{cases} 0 & j \le i - 2 \\ i\, p_o(1 - p_o)^{i-1}(1 - \sigma)^{M-i} & j = i - 1 \\ (1-p_o)^{i}(M-i)\sigma(1-\sigma)^{M-i-1} + [1-ip_o(1-p_o)^{i-1}](1-\sigma)^{M-i} & j = i \\ [1 - (1 - p_o)^{i}](M - i)\sigma(1 - \sigma)^{M-i-1} & j = i + 1 \\ \binom{M - i}{j - i} \sigma^{j-i} (1 - \sigma)^{M-j} & j \ge i + 2 \end{cases} \tag{6.38}$$

$$p_{ij}(ac) = \begin{cases} 0 & j \le i - 2 \\ i\, p_c(1 - p_c)^{i-1}(1 - \sigma)^{M-i} & j = i - 1 \\ (1-p_c)^{i}(M-i)\sigma(1-\sigma)^{M-i-1} + [1-ip_c(1-p_c)^{i-1}](1-\sigma)^{M-i} & j = i \\ [1 - (1 - p_c)^{i}](M - i)\sigma(1 - \sigma)^{M-i-1} & j = i + 1 \\ \binom{M - i}{j - i}\sigma^{j-i}(1 - \sigma)^{M-j} & j \ge i + 2 \end{cases} \tag{6.39}$$

$$p_{ij}(ro) = \begin{cases} i\,p_o(1 - p_o)^{i-1} & j = i - 1 \\ 1 - i\,p_o(1 - p_o)^{i-1} & j = i \\ 0 & \text{otherwise} \end{cases} \qquad (6.40)$$

$$p_{ij}(rc) = \begin{cases} i\,p_c(1 - p_c)^{i-1} & j = i - 1 \\ 1 - i\,p_c(1 - p_c)^{i-1} & j = i \\ 0 & \text{otherwise} \end{cases} \qquad (6.41)$$

As before, we neglect the uninteresting cases when σ, p_o or $p_c = 0$.

Average packet delay and channel throughput rate

Consider a stationary control policy $f : S \to A$ uniquely specified by the nonintersecting sets S_{ao}, S_{ac}, S_{ro} and S_{rc} such that

$$S = S_{ao} \cup S_{ac} \cup S_{ro} \cup S_{rc}$$

and

$$f(i) = \begin{cases} ao & i \in S_{ao} \\ ac & i \in S_{ac} \\ ro & i \in S_{ro} \\ rc & i \in S_{rc} \end{cases} \qquad (6.42)$$

Let

$$S_a = S_{ao} \cup S_{ac}$$
$$S_r = S_{ro} \cup S_{rc} \tag{6.43}$$

Define the expected immediate (delay) costs to be

$$C_i(ao) = C_i(ac) = i \tag{6.44}$$

and

$$C_i(ro) = C_i(rc) = i + (M - i)\sigma \cdot d_r = M \tag{6.45}$$

The cost rate of the process N^t under policy f is given by Eq. (6.24)

$$g_d(f) \quad \sum_{i=0}^{M} i\, \pi_i(f) + d_r \sum_{i \epsilon S_r} (M - i)\sigma\, \pi_i(f) \tag{6.24}$$

and the average packet delay (including rejection delay) is given by Eq. (6.27)

$$D = \frac{g_d(f)}{S_{out}} + R + 1 \tag{6.27}$$

To obtain S_{out}, the following expected immediate (throughput) costs are adopted.

$$C_i(ao) = - S_{out}(i, ao)$$

$$= -[i\, p_o(1-p_o)^{i-1}(1-\sigma)^{M-i} + (1-p_o)^{i}(M-i)(1-\sigma)^{M-i-1}] \tag{6.46}$$

$$C_i(ac) = - S_{out}(i, ac)$$

$$= -[i\, p_c(1-p_c)^{i-1}(1-\sigma)^{M-i} + (1-p_c)^{i}(M-i)\sigma(1-\sigma)^{M-i-1}] \tag{6.47}$$

$$C_i(ro) = - S_{out}(i, ro)$$

$$= -i\, p_o(1 - p_o)^{i-1} \tag{6.48}$$

$$C_i(rc) = - S_{out}(i, rc)$$

$$= -i\, p_c(1 - p_c)^{i-1} \tag{6.49}$$

The cost rate of the Markov process and the stationary channel throughput rate are again given by

$$g_s(f) = - \sum_{i=0}^{M} \pi_i(f)\, S_{out}(i,f)$$

and

$$S_{out} = -g_s(f) \tag{6.50}$$

Thus,

$$D = - \frac{g_d(f)}{g_s(f)} + R + 1 \tag{6.51}$$

6.4 A Theorem on the Equivalence of the Performance Measures

The channel throughput rate S_{out} and average packet delay D constitute the performance measures of interest for the controlled channel. Under any one of the previously described channel control procedures and given a stationary control policy f, either one of the performance measures can be evaluated by appropriate definitions of the state transition probabilities and expected immediate costs of the Markov decision process N^t. The value-determination operation yields the cost rate of N^t, from which the value of the performance measure can be computed. Given a single performance measure, the policy-iteration method will, in fact, lead to an optimal stationary policy with respect to the given performance measure in a finite number of steps.

Under any one of the control procedures, some obvious optimization problems seem to be:

(1) $\underset{f \in P_s}{\text{Min}}\ D$ given some (minimum) constraint on S_{out}

(2) $\underset{f \in P_s}{\text{Max}}\ S_{out}$ given some (maximum) constraint on D

(3) $\underset{f \in P_s}{\text{Min}}\ (D - \beta S_{out})$ for some $\beta > 0$

where P_s is the class of all stationary policies. Markov decision theory as introduced in Section 6.2 does not provide for the solution of the first two optimization problems with constraints. In the third problem, there is no natural candidate for the positive constant β

which determines the relative weights we put on the two performance measures (D and S_{out}). Luckily, we have been able to establish the following lemma and theorem which enable us to get around this difficulty.

Lemma 6.3 Under each of the control procedures ICP, RCP or IRCP

$$g_d(f) = \frac{g_s(f)}{\sigma} + M \tag{6.50}$$

where f is any stationary control policy.

Proof The proof hinges on the observation that under a stationary control policy, N^t is a finite-state Markov process with stationary transition probabilities in which case the stationary channel throughput rate S_{out} must be equal to the stationary channel input rate.

We first consider the input control procedure (ICP). From Eqs. (6.21) and (6.24)

$$g_d(f) = \sum_{i=0}^{M} i\,\pi_i(f) + \frac{1}{\sigma} \sum_{i\in S_r} (M - i)\sigma\,\pi_i(f)$$

$$= \sum_{i=0}^{M} i\,\pi_i(f) + \frac{1}{\sigma} \sum_{i\in S_r} (M - i)\sigma\,\pi_i(f)$$

$$+ \frac{1}{\sigma} \sum_{i\in S_a} (M - i)\sigma\,\pi_i(f) - \frac{1}{\sigma} \sum_{i\in S_a} (M - i)\sigma\,\pi_i(f)$$

$$= \sum_{i=0}^{M} i\,\pi_i(f) + M - \sum_{i=0}^{M} i\,\pi_i(f) - \frac{1}{\sigma} \sum_{i\in S_a} (M-i)\sigma\,\pi_i(f)$$

$$= -\frac{i}{\sigma} \sum_{i\in S_a} (M - i)\sigma\,\pi_i(f) + M$$

Note that $\sum_{i\in S_a} (M - i)\sigma\,\pi_i(f)$ is just the stationary channel input rate and is thus equal to the stationary channel throughput rate $S_{out} = -g_s(f)$. Hence,

$$g_d(f) = \frac{g_s(f)}{\sigma} + 1$$

and the proof is complete for ICP.

We next consider the retransmission control procedure (RCP). From Eq. (6.35),

$$g_d(f) = \sum_{i=0}^{M} i\,\pi_i(f)$$

$$= \sum_{i=0}^{M} (i - M)\,\pi_i(f) + M$$

$$= -\frac{1}{\sigma} \sum_{i=0}^{M} (M - i)\sigma\,\pi_i(f) + M$$

Again, $\sum_{i=0}^{M} (M - i)\sigma\, \pi_i(f)$ is just the stationary channel input rate and is thus equal to $S_{out} = -g_s(f)$. Thus,

$$g_d(f) = \frac{g_s(f)}{\sigma} + M$$

and the proof is complete for RCP.

The proof for IRCP is identical to that for ICP.

Q.E.D.

Theorem 6.4 Under each of the control procedures ICP, RCP or IRCP,

(i) there exists a stationary policy $\hat{f}$ such that

$$g_d(\hat{f}) = \min_{f \in P_s} g_d(f)$$

if and only if

$$g_s(\hat{f}) = \min_{f \in P_s} g_s(f)$$

(ii) if $\hat{f}$ is a stationary policy satisfying the preceding condition, then $\hat{f}$ minimizes D over the class P of all policies and at the same time, $\hat{f}$ maximizes S_{out} over the class P of all policies.

Proof (i) This is a direct consequence of Lemma 6.3 and the existence of $\hat{f}$ is guaranteed by Theorem 6.2. (ii) By Eqs. (6.30) and (6.31), $\hat{f}$ minimizes D and maximizes S_{out} over

the class of all stationary policies. The generalization to the class P of all policies is a consequence of Theorem 6.1. Q.E.D.

Lemma 6.3 and Theorem 6.4 can be generalized to control procedures similar to ICP, RCP and IRCP, but with more alternatives in their action spaces. This is done in Appendix F.

Summarizing the results in Theorems 6.1, 6.2 and 6.4, we state that under each of the control procedures ICP, RCP and IRCP, a stationary policy $f : S \to A$ always exists which minimizes the average packet delay D and maximizes the stationary channel throughput rate S_{out} over the class P of all policies. Such an optimal control policy and its channel performance measures D and S_{out} can be obtained by applying the policy-iteration method. In the next section, we shall present an efficient computational algorithm which utilizes the policy-iteration method.

An interpretation of Theorem 6.4 and the optimization problem

The average packet delay D is given by Eq. (6.31) as

$$D = -\frac{g_d(f)}{g_s(f)} + R + 1 \tag{6.31}$$

where f is a stationary control policy in any of the above control procedures. Applying Eqs. (6.30) and (6.50) to substitute for $g_d(f)$ and $g_s(f)$ in the above equation, we have

$$D = R + 1 + \left(\frac{M}{S_{out}} - \frac{1}{\mu} \right) \tag{6.51}$$

which relates D as a one-to-one function of S_{out} given fixed values of R, M and σ. (Note that the last two variables determine the channel load line.) Moreover, this function is monotonically decreasing.

Assuming a fixed R, we show in Fig. 6-6 a family of curves each of which depicts D as a function of S_{out} given by Eq. (6.51). The parameters M and σ, which determine the channel load line, also define a curve in the two-dimensional space of the performance measures D and S_{out}. We may consider each one of the control procedures in Section 6.3 as a mathematical operator which maps P_s (the space of all stationary policies) into the above curve. Each f in P_s is mapped into one point on the curve. The range space of the operator must be a proper subset of points on the curve. Otherwise, it is possible that $D = R + 1$ and $S_{out} = M\sigma$ (i.e., no congestion at all!). The optimization problem thus corresponds to finding the extreme points (maximum S_{out} and minimum D) of the range space. Since the curve under consideration is monotonically decreasing, these extreme points coincide. Thus, the same control policy f must maximize S_{out} and minimize D at the same time.

Given a family of channel load lines (e.g., M varying from 0 to ∞ at fixed σ or σ varying from 0 to 1 at fixed M) each channel control procedure gives rise to an infeasible region such as shown in Fig. 6-6. The boundary of this region represents the optimum throughput-delay tradeoff under the above constraints. The optimization problem here is to find the optimal control policies which

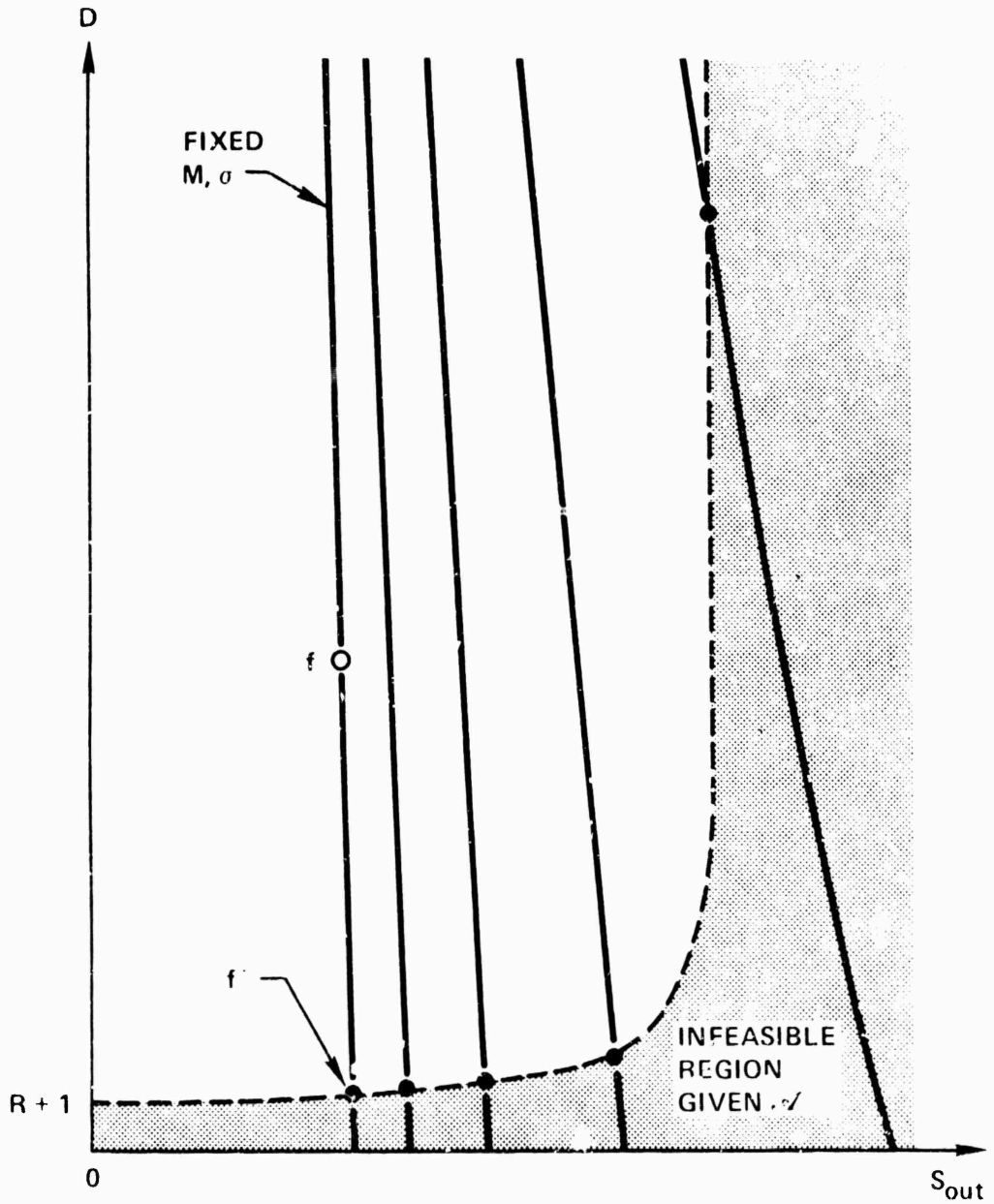


Figure 6-6. Optimum Performance of a Channel Control Procedure.

achieve this optimum channel performance. Below we give a computational algorithm to do so.

6.5 An Efficient Computational Algorithm (POLITE)

In any optimization problem, the optimum solution is readily available if we can enumerate all possible solutions. Thus, an optimization problem is solvable in the sense that the task of enumerating the set of possible solutions is within the limits of the computing capability of our machine(s). Even when a problem is solvable, we must look for ways to reduce the computational cost in terms of the time and space allocation of our machine(s) to the problem.

For the problem at hand, we have seen the tremendous savings in computational cost by reducing the set of possible solutions from the class of all policies to the class of stationary policies. Still, we have not altogether escaped from the "curse of dimensionality" since, for example, if S has 300 states and A has two actions, there are still 2^{300} (an astronomical number) stationary policies to consider. Howard's policy-iteration method described in Section 6.2.3 enables us to find an optimal policy usually in a small number of iterations. The method is composed of two parts as shown in Fig. 6-1, the value-determination operation and the policy-improvement routine. The difficulty now arises in the solution of the $(M + 1)$ linear simultaneous equations in Eqs. (6.10) for g and the relative values of v_i (setting $v_0 = 0$) when M is large (say, a few hundred, which is our range of interest).

$$g + v_i = C_i + \sum_{j=0}^{M} p_{ij} v_j \qquad i = 0, 1, 2, \ldots M \qquad (6.10)$$

For example, if $M = 400$, the task to solve Eqs. (6.10) is somewhat equivalent to inverting a 401 x 401 matrix with 160,801 entries! The fact that the state transition probabilities $p_{ij} = 0$ for $j \leq i - 2$ in all our models enables us to decompose the $(M + 1)$ linear simultaneous equations in Eqs. (6.10) into two sets of M linear simultaneous equations, each of which can then be solved by applying Algorithm 5.1. We summarize the procedure in the following algorithm, which plays a crucial role in making possible the use of the policy-iteration method to solve optimization problems involving hundreds of channel users. Its derivation is given in Appendix E.

Algorithm 6.5

This algorithm solves for g and $\{v_i\}_{i=1}^{M}$ in the following set of $(M + 1)$ linear simultaneous equations,

$$\begin{aligned} g &= C_0 + \sum_{j=1}^{M} p_{0j} v_j \\ g + v_1 &= C_1 + \sum_{j=1}^{M} p_{1j} v_j \\ g + v_i &= C_i + \sum_{j=i-1}^{M} p_{ij} v_j \qquad i = 2, 3, \ldots, M \end{aligned} \qquad (6.52)$$

where

$$\sum_{j=0}^{M} p_{0j} = \sum_{j=i-1}^{M} p_{ij} = 1 \qquad i = 1, 2, \ldots, M$$

(1) Define

$$b_{M-1} = \frac{1}{p_{M,M-1}}$$

$$d_{M-1} = -\frac{C_M}{p_{M,M-1}}$$

(2) For $i = M - 1, M - 2, \ldots, 2$ solve recursively

$$b_{i-1} = \frac{1}{p_{i,i-1}} \left[b_i + 1 - \sum_{j=i}^{M-1} p_{ij}\, b_j \right]$$

$$d_{i-1} = \frac{1}{p_{i,i-1}} \left[d_i - C_i - \sum_{j=i}^{M-1} p_{ij}\, d_j \right]$$

(3) Define

$$u_M = -\frac{1}{p_{10}} \left[b_1 + 1 - \sum_{j=1}^{M-1} p_{1j}\, b_j \right]$$

$$w_M = -\frac{1}{p_{10}} \left[d_1 - C_1 - \sum_{j=1}^{M-1} p_{1j}\, d_j \right]$$

$$u_i = u_M + b_i \qquad i = 1, 2, \ldots, M - 1$$

$$w_i = w_M + d_i$$

(4) Let

$$g = \frac{C_0 + \sum_{j=1}^{M} p_{0j} w_j}{1 - \sum_{j=1}^{M} p_{0j} u_j}$$

$$v_i = u_i g + w_i \qquad i = 1, 2, \ldots, M$$

Algorithm 6.5 has the same advantages as Algorithm 5.1 (which it utilizes) discussed in Section 5.3.1. Briefly, they are:

(1) The crucial variables b_i and d_i in the algorithm are computed recursively such that the state transition probabilities p_{ij} can be computed as needed. This eliminates the need for storing the $\frac{(M + 1)(M + 2)}{2} + M$ elements in the state transition matrix and virtually eliminates any machine storage constraint on the dimensionality of the optimization problem.

(2) The number of arithmetic operations required is also smaller than that of a standard solution method such as Gauss elimination [CRAI 64].

These considerations render the policy-iteration method a very efficient tool in the solution of our optimization problem.

We give below an algorithm (called POLITE) which combines the POLicy-ITEration method, Algorithm 6.5 and Theorem 6.4. Given a Markov decision process model of the channel, POLITE finds the optimal control policy and evaluates the optimum channel performance measures.

Algorithm 6.6 (POLITE)

Given the Markov decision process N^t with

state space $S = \{0, 1, 2, \ldots, M\}$,

finite action space A (ICP, RCP or IRCP),

throughput or delay costs $\{C_i(a) \mid i \in S, a \in A\}$,

state transition probabilities $\{p_{ij}(a) \mid i, j \in S, a \in A, p_{ij}(a) = 0 \text{ if } j \leq i - 2\}$,

and stationary policies $f : S \to A$.

To determine a stationary policy f^* such that the cost rate g of N^t is minimized.

Start at either step (1) or step (2).

(1) Given a policy f, apply Algorithm 6.5 to obtain g and $\{v_i\}_{i=1}^{M}$; $p_{ij}(f)$ and $C_i(f)$ are computed when need in Algorithm 6.5.

(2) Given a set of $\{v_i\}_{i=1}^{M}$ for state $i = 0, 1, \ldots, M$ define the test quantity

$$\text{Cost}(i, a) = C_i(a) + \sum_{j=1}^{M} p_{ij}(a)v_j \tag{6.53}$$

Find $\hat{a}$ such that $\text{Cost}(i, \hat{a}) = \min_{a \in A} \text{Cost}(i, a)$.

If $\text{Cost}(i, f(i)) = \text{Cost}(i, \hat{a})$, then let $\hat{f}(i) = f(i)$; otherwise, let $\hat{f}(i) = \hat{a}$.

(3) If $\hat{f}$ and f are identical, go to step (5).

(4) Replace f by $\hat{f}$ and go to step (1).

(5) $f^* = f$ is an optimal control policy.

(6) $g = g_s(f^*)$ or $g_d(f^*)$ depending on the expected immediate costs $C_i(a)$.

Apply $g_d(f) = \frac{g_s(f)}{\sigma} + M$

(7) The optimum performance measures are,

$$S^*_{out} = - g_s(f^*)$$

$$D^* = - \frac{g_d(f^*)}{g_s(f^*)} + R + 1$$

6.6 Evaluation of Control Procedures by POLITE

6.6.1 Computational Costs and Convergence

The POLITE algorithm is our tool for computing optimal control policies and evaluating performance measures of the controlled channel using ICP, RCP or IRCP. The algorithm has been coded in Fortran and runs on the IBM 360/91 of the UCLA Campus Computing Network (CCN). For the numerical examples we considered, which will be given in the following sections, the core memory requirement is less than 90K bytes and the job CPU time for each run is between 1 to 6 seconds. (Double precision is used, M is up to 508 and the number of algorithm iterations* is in all cases less than 5.) These numbers translate to less than one dollar per run on the average at the current CCN charge rate and are very reasonable considering the size of the problems involved. For comparison, consider the following example. If M = 400, the state transition matrix $[p_{ij}]$ alone has $\frac{(401)(402)}{2} + 400 = 81001$ nonzero entries and requires 649K bytes of memory to store it in double precision.

---

*By an iteration of the algorithm POLITE, we mean a complete cycle of steps (1) to (4) in the algorithm.

No conscious effort has been made to optimize the program code except for the following options. First, in step (2) of POLITE, when Cost(i, f(i)) and Cost(i, $\hat{a}$) are compared, they are assumed to be equal if Cost(i, $\hat{a}$) is within $1 \pm \varepsilon$ of Cost(i, f(i)). In all our numerical computations, ε is taken to be 10^{-5}. Second, to prevent the occurrence of "underflows" during program execution, some threshold must be specified in the program so that whenever a number is less than the threshold it is put equal to zero. For our purpos., the threshold value is taken to be 10^{-30} (instead of the possible 10^{-75} in the IBM 360/91) to save some comput.ions. Smaller threshold values have been used to recompute several cases. No discrepancy in the program output values is observed.

In applying POLITE to solve the ICP and RCP optimization problems, we adopt the following strategy. A control limit policy is always used as the initial control policy to start the algorithm at step (1). This control limit is chosen somewhere between the operating point n_o on the channel load line and the unstable equilibrium point n_c (see Fig. 5-6(b)). Under such an initial control policy, the algorithm requires in most cases between 2 to 4 iterations to arrive at the optimal control policy (algorithm termination).

Although our optimization problem can now be solved by POLITE with relatively small time-space demands on the computer, there exists another constraint which bounds the dimensionality of our problem--the precision of numbers in the computer. When M is large and/or A has many elements, we need to distinguish numbers which are so close

together that they are no longer distinguishable given the precision of the computer. Furthermore, increases in the number of recursive steps within the algorithm produce bigger round-off errors, the effect of which is becoming more pronounced. We found that for a value of M larger than 500, the program may not converge* if the initial control policy is not close to the optimal policy. This is (probably) caused by the accumulation of round-off errors as the algorithm requires more iterations for an initial policy which is farther away from the optimal policy.

6.6.2 "Optimality" of the Control Limit Policy

Consider ICP and RCP. The action space A of both control procedures consists of two actions $\{a_o, a_c\}$. a_o is the operating action, designed to give good channel throughput-delay performance conditioning on equilibrium conditions. a_o corresponds to "accept" in ICP and p_o (or K_o) in RCP. a_c is the control action, designed to prevent the channel from going into saturation. a_c corresponds to "reject" in ICP and p_c (or K_c) in RCP.

Our intuition suggests that a good control policy (for either ICP or RCP) must be such that the control action should be applied whenever the channel backlog size N^t exceeds some threshold value to prevent it from drifting toward saturation. But as soon as N^t decreases below this threshold value, the control action should be

---

*In our computations, each application of POLITE is allowed a maximum of 5 iterations, after which the program stops. Remember that the algorithm is guaranteed to terminate by Theorem 5.2. The difficulty here stems from machine limitations rather than the algorithm itself.

replaced by the operating action, since its use costs the system much more in terms of both channel throughput and packet delay. This intuition has been confirmed in all our numerical computations for ICP and RCP. In each case, the optimal control policy given by POLITE is a control limit policy of the following form.

$$f(i) = \begin{cases} a_o & i \leq \hat{n} \\ a_c & i > \hat{n} \end{cases} \tag{6.54}$$

where $\hat{n}$ is said to be the control limit (CL) of the control limit policy f.

A rigorous mathematical proof of the optimality of the control limit policy remains an open problem. In many problems characterized by optimal policies of the CL type, the usual method of attack in their proof is to demonstrate monotonicity for the sequences $\{v_i\}$ and $\{\text{Cost}(i, a_o) - \text{Cost}(i, a_c)\}$. The lack of monotonicy in most such sequences is clearly seen in Figs. 6-7 to 6-10. These figures also serve to illustrate some of the steps of the algorithm POLITE.

An ICP example is shown in Figs. 6-7 and 6-8 where the sequences $\{\text{Cost}(i, a) - \text{Cost}(i, r)\}$ and $\{v_i\}$ have been plotted as functions of i. Delay costs corresponding to Eqs. (6.22) and (6.23) are assumed. Each curve in these figures is obtained using the control policy generated during the previous iteration of the algorithm. Consider Fig. 6-7. The initial control policy is a control limit policy with $\hat{n} = 40$ (which interestingly corresponds to the joining point of

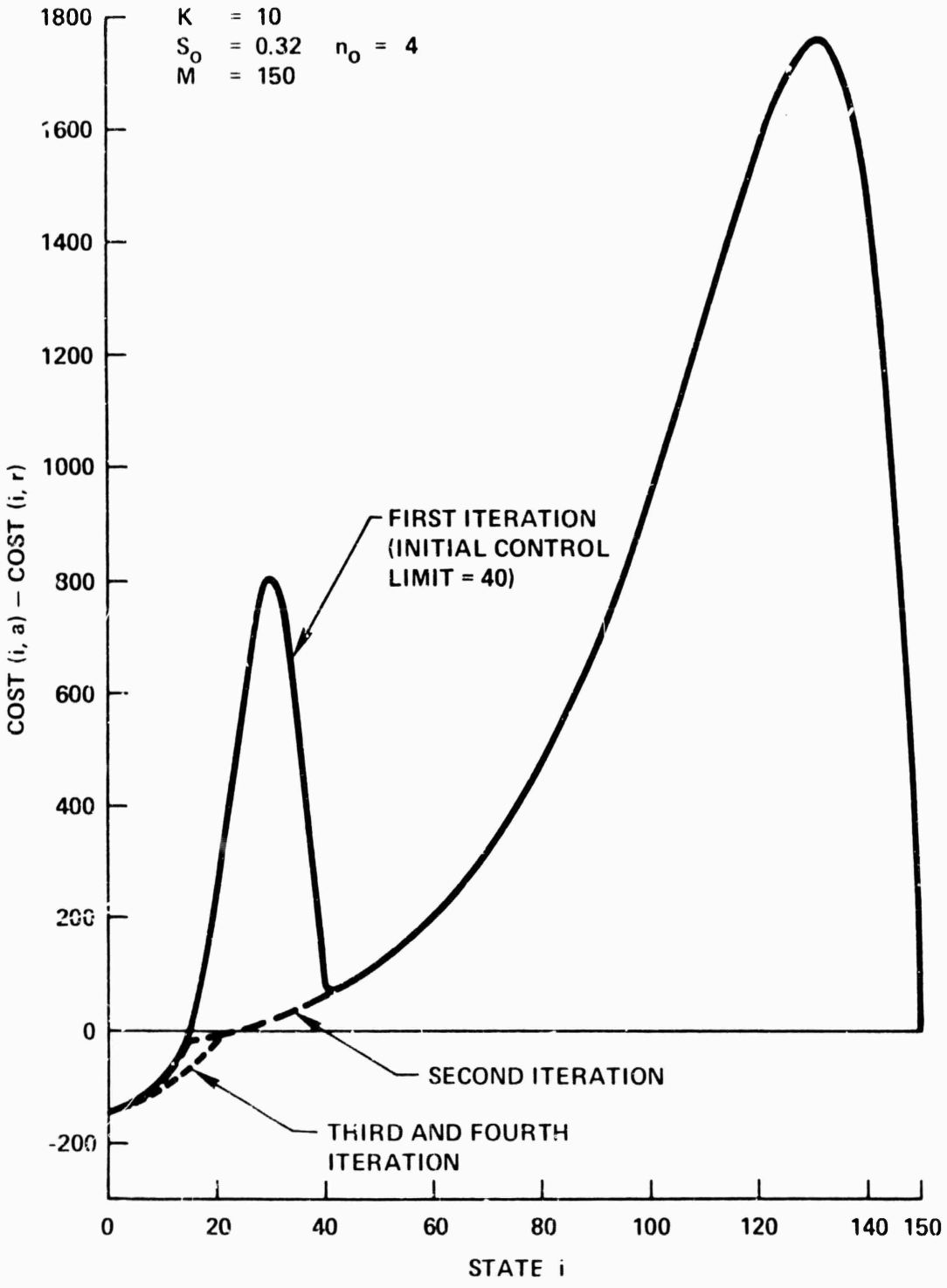


Figure 6-7. POLITE Iterations for ICP With Delay Costs – Control Limits.

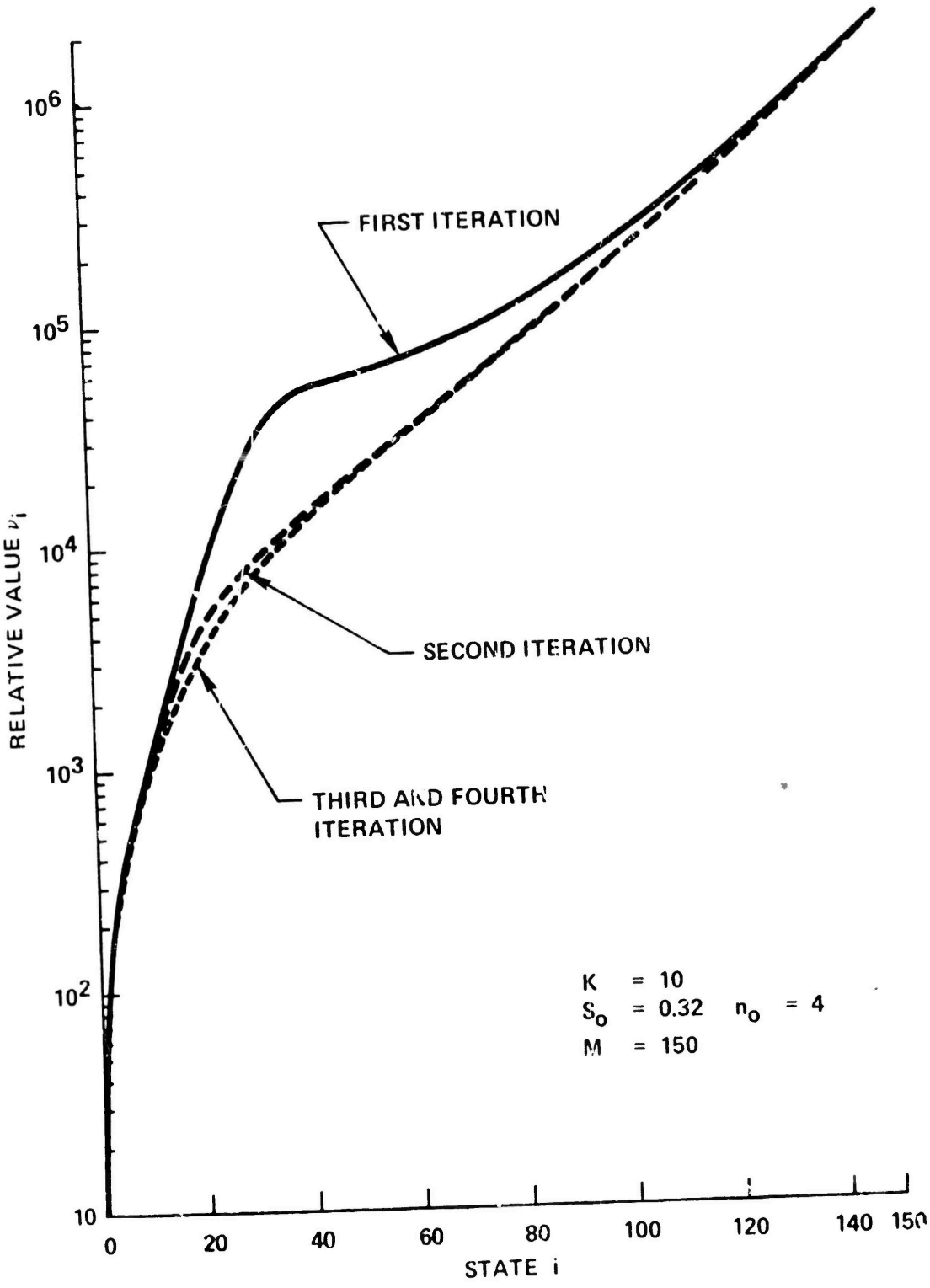


Figure 6-8. POLITE Iterations for ICP With Delay Costs – ν_i

the two humps of the first iteration curve). The first iteration curve crosses zero exactly once between $i = 14$ and $i = 15$. Thus, $\hat{n} = 14$ becomes the control policy for the second iteration. (Recall step (2) in POLITE.) The second iteration curve yields the control policy $\hat{n} = 23$. Finally, the optimal control policy ($\hat{n} = 22$) is obtained in both the third and fourth iterations and the algorithm terminates. In Fig. 6-8, the relative values v_i in each iteration are shown. We see that v_i is monotonically increasing in i. This implies that the expected total cost in delay(over a finite time horizon) increases as a function of the channel state i at time zero (see Eq. (6.7)).

A RCP example is shown in Figs. 6-9 and 6-10. Throughput costs corresponding to Eqs. (6.36) and (6.37) are assumed (which explains the negative values in Fig. 6-10). Note that the algorithm terminates in only three iterations.

Observe in Figs. 6-7 and 6-9 that when the initial control policy for POLITE is a CL policy, not only is the final optimal policy a CL policy, but all intermediate control policies generated by POLITE are of the control limit type. To test if POLITE generates CL policies only when a CL policy is fed into the algorithm as the initial policy, we tried the following. Let $0 = m_1 < m_2 < \ldots < m_J = M$. Define the control policy

$$f'(i) = \begin{cases} a_o & i = 0 \text{ or } m_j < i \leq m_{j+1}, \ j \text{ is odd} \\ a_c & m_j < i \leq m_{j+1}, \ j \text{ is even} \end{cases}$$

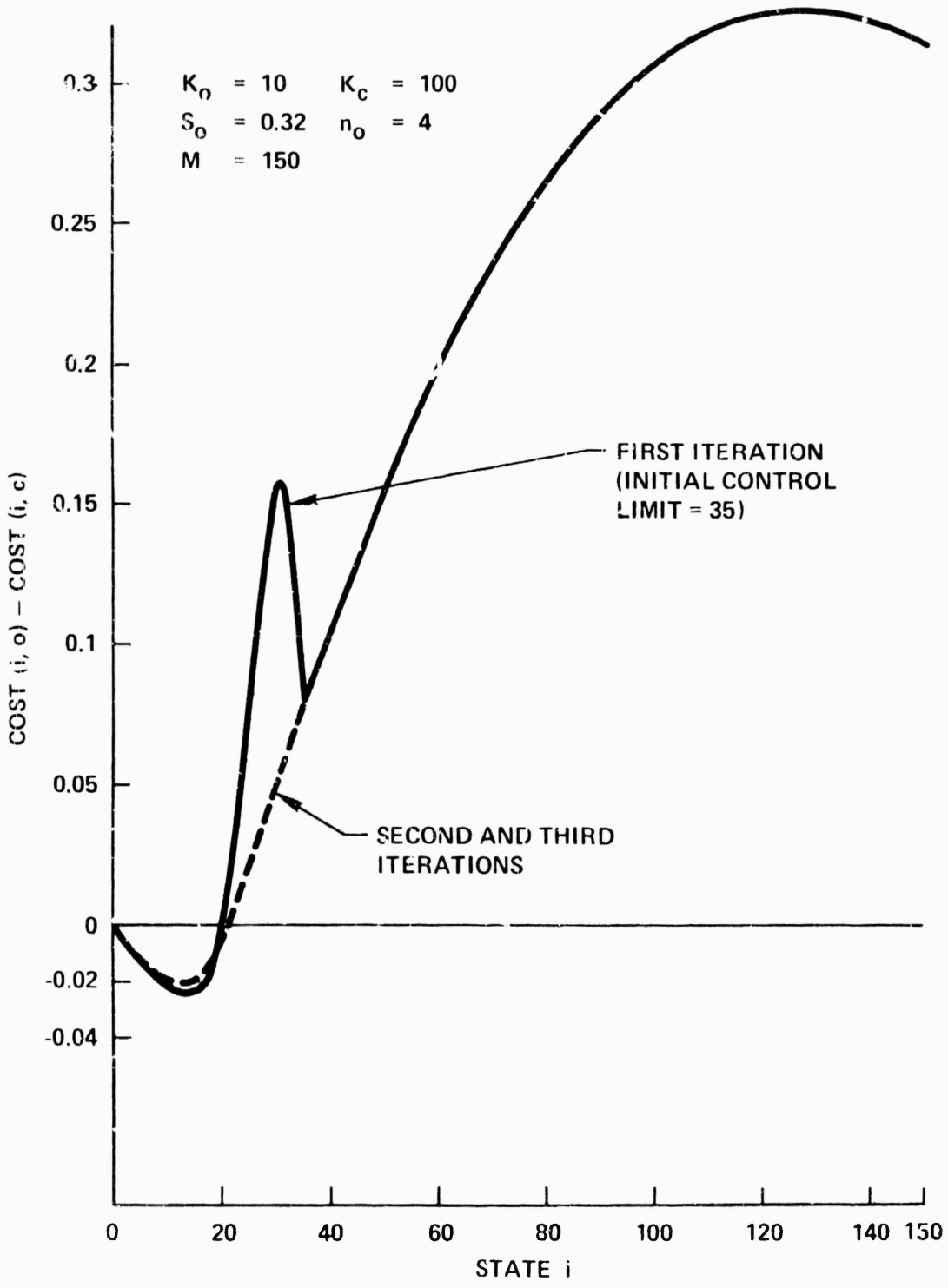


Figure 6-9. POLITE Iterations for RCP With Throughput Costs – Control Limits.

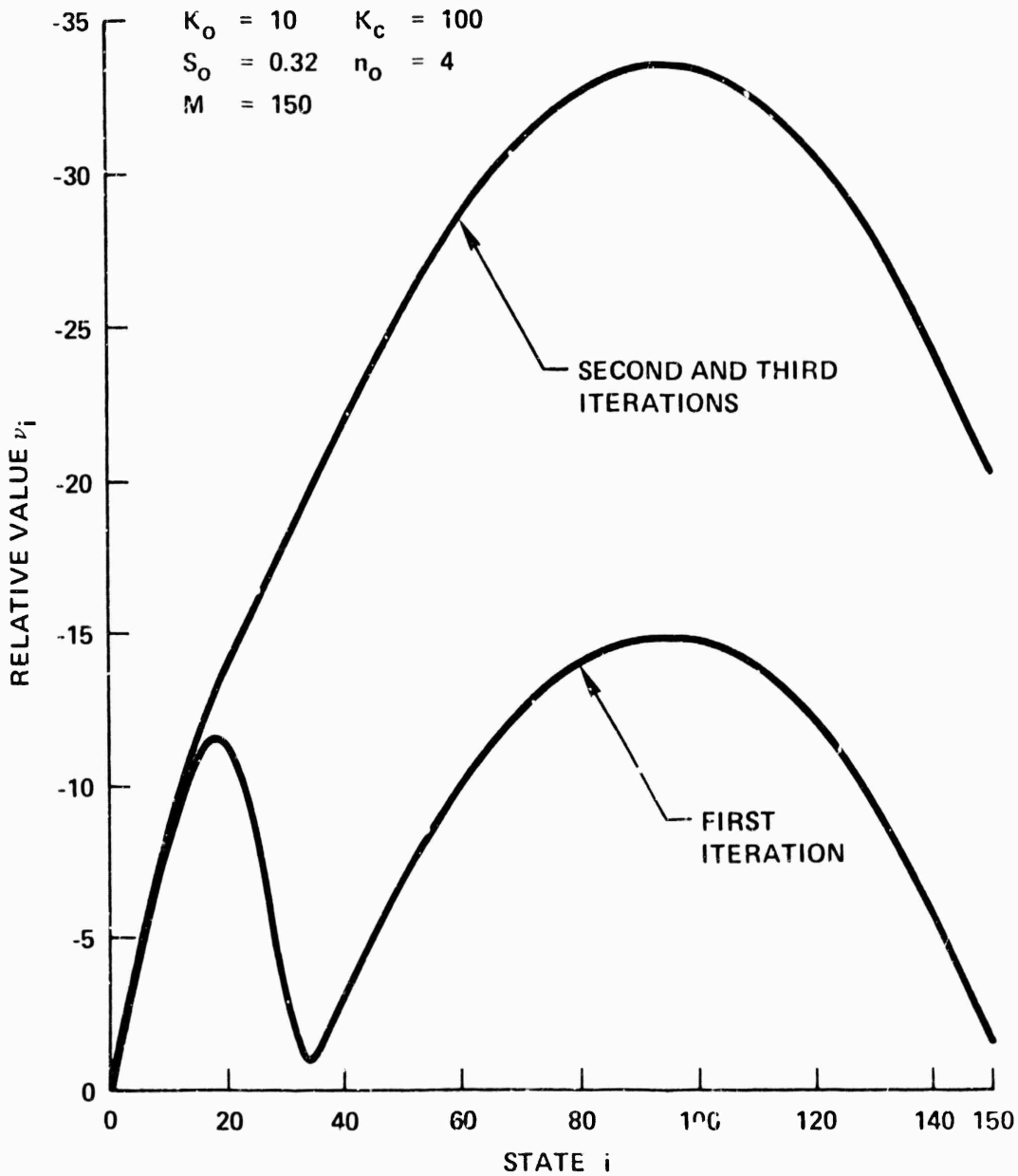


Figure 6-10. POLITE Iterations for RCP With Throughput Costs – v_i

Such a control policy was used as the initial policy to drive the algorithm POLITE in several cases. In each case, the same CL policy as before was generated by POLITE to be the optimal policy.

6.6.3 Channel Performance

We show in this section throughput-delay performances of the controlled channel using ICP, RCP or IRCP.

Given an unstable channel load line, the throughput-delay performance at the operating point (n_o, S_o) is what we strive to achieve through dynamic channel control. Thus, it is essential that the operating value of K gives an operating point (n_o, S_o) close to the optimum. In Figs. 3-4 to 3-5, we see that $K = 10$ is an excellent choice and will be used throughout this chapter as the operating value of K. The channel load line is a straight line uniquely specified by its intercept on the vertical axis, M, and its slope $-\frac{1}{\sigma}$. However, often we would prefer to specify the load line by specifying M and the operating point (n_o, S_o) on the equilibrium contour (instead of σ). Thus, different load lines specified by the same channel operating point can be compared by showing how well they approach the throughput-delay performance at the operating point.

The equilibrium contour corresponding to $K = 10$ is shown in Fig. 5-3. Each channel load line to be used in our computations will be specified by M and one other point on the (n,S) plane. The points shown in Table 6.1 will be used.

n_o =	1	2	3	4	5	7	10
S_o =	0.2	0.25	0.3	0.32	0.34	0.36	0.374

Table 6.1 Points on the $K = 10$ contour.

(Note that these points only approximate points on the $K = 10$ contour. For example, given $S_o = 0.32$, n_o given by the $K = 10$ contour is actually between 3 and 4, but has been rounded off to 4 for convenience.) In particular, the points $(n_o, S_o) = (4, 0.32)$ and $(7, 0.36)$ will be used in most of our examples. Assuming a large M, these points correspond to a channel which is moderately to very heavily "loaded" when the problems of channel instability and channel control become significant.

From our discussion in the last section, all control policies considered below for ICP and RCP are of the CL type.

In ICP the control action is to reject all new packet arrivals. In RCP the control action is to use a large enough value of $K = K_c$ which renders the channel load line stable. We illustrate this last statement in Fig. 6-11. The average packet delay D given by an optimal RCP control policy is shown as a function of K_c. Note that K_c somewhat less than the necessary value of K to render the channel load line stable can be used. However, if K_c is too small, the channel performance "blows up" since now the controlled channel is still unstable. Observe that for a sufficiently large K_c, D is quite insensitive to its exact value except when $S_o = 0.36$, in which case D increases slowly with K_c. Note that for the same S_o, a much larger K_c is

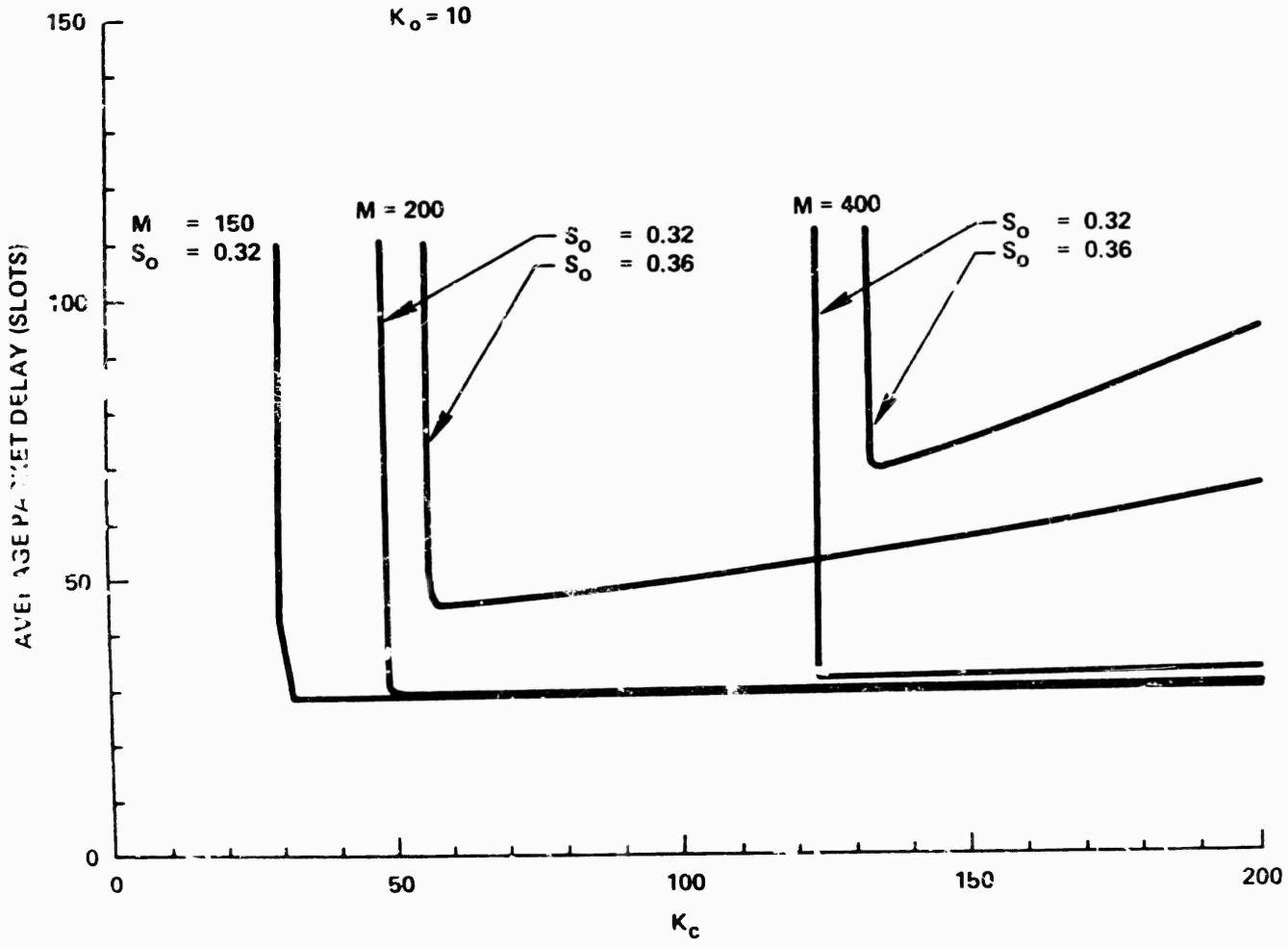


Figure 6.11. RCP Channel Performance Versus K_c.

required for a larger M. In the limit as $M \to \infty$, RCP becomes ineffective since no sufficiently large value of K can be used for K_c.

Knowing that the optimal control policy is a CL policy, we show in Figs. 6-12 to 6-15, the channel performance measures S_{out} and D (given by ICP and RCP for $M = 200,400$ and $S_o = 0.32, 0.36$) over a range of control limits. Observe that the same control limit minimizes D and maximizes S_{out} at the same time as predicted by Theorem 6.4. Note the amazing flatness of S_{out} and D near the optimum point, especially when $S_o = 0.32$ and $M = 200$ in Figs. 6-12 and 6-13. The consequence is that even if a nonoptimal control policy is used (due to, for example, not knowing the exact current backlog size such as in most practical systems), it is still possible to achieve a throughput-delay performance close to the optimum. However, such flatness of S_{out} and D is not as pronounced when S_o is 0.36. Comparing the four figures, we see that the optimum values of S_{out} and D given by ICP and RCP are approximately the same, but RCP gives less severe degradation in channel performance with control limits much smaller and much larger than the optimal. However, recall from Fig. 6-11 the potential disastrous channel behavior if K_c is not sufficiently large. This must be taken into consideration in any system design using RCP since in a practical system both the parameters M and σ may change with time. To provide the necessary design safety margin, a much bigger value of K_c than deemed necessary may have to be adopted. In Fig. 6-13, we show the degradation in channel performance when $K_c = 200$ is used instead of $K_c = 60$. (The use of $K_c = 200$ allows the channel to support more than 400 users instead of 200.) On the other hand, M

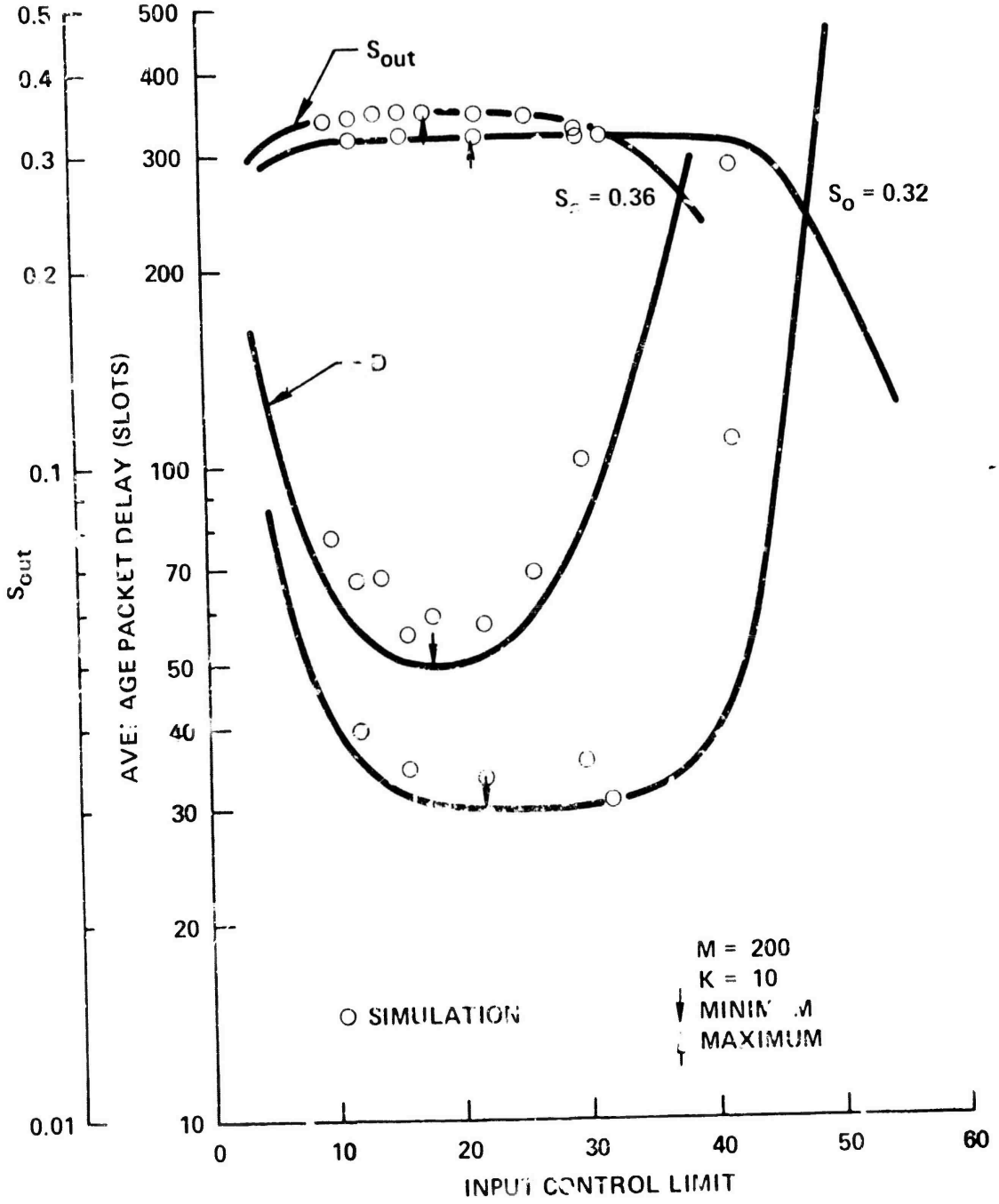


Figure 6-12. Channel Performance Versus ICP Control Limit for M = 200.

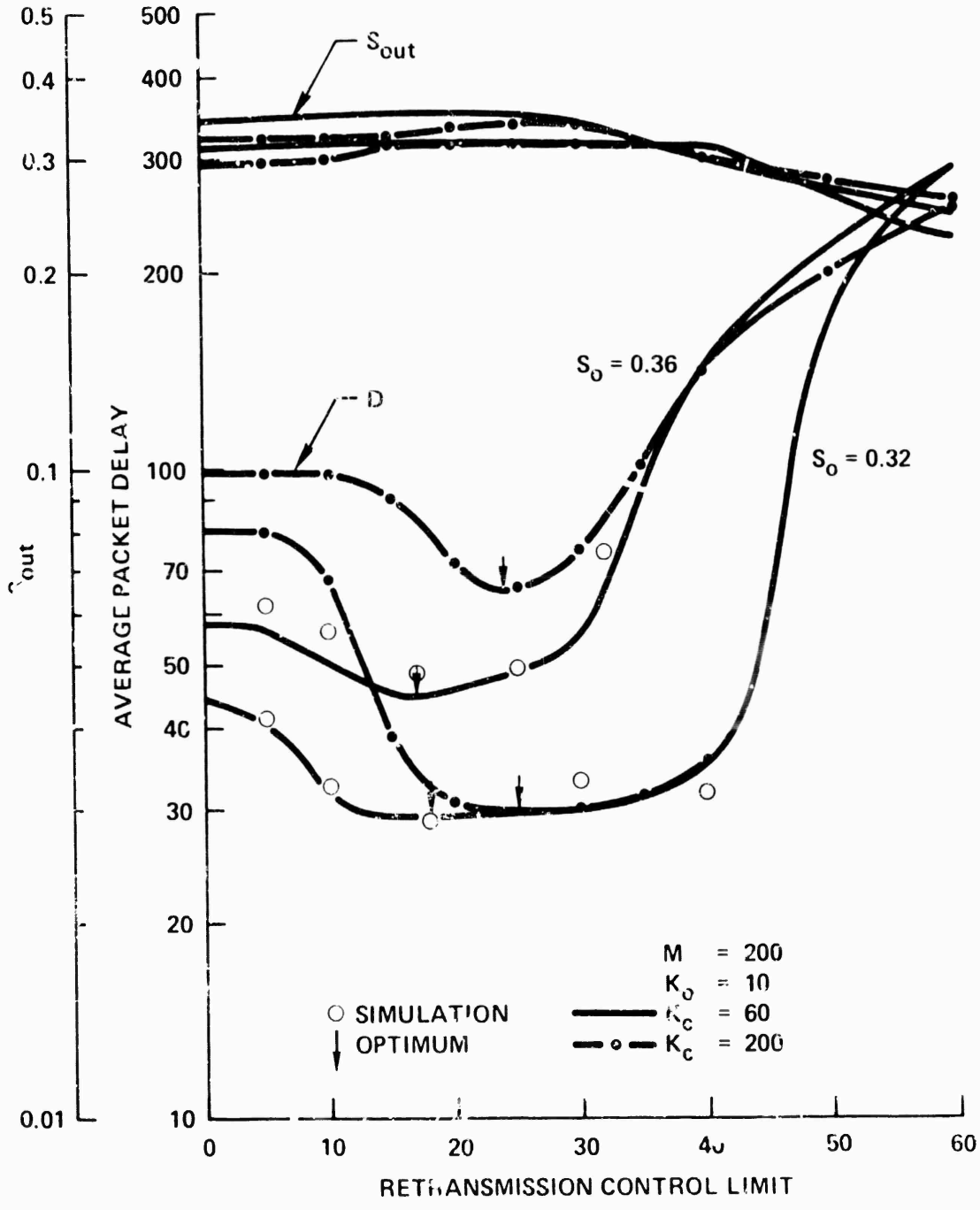


Figure G-13. Channel Performance Versus RCP Control Limit for M = 200

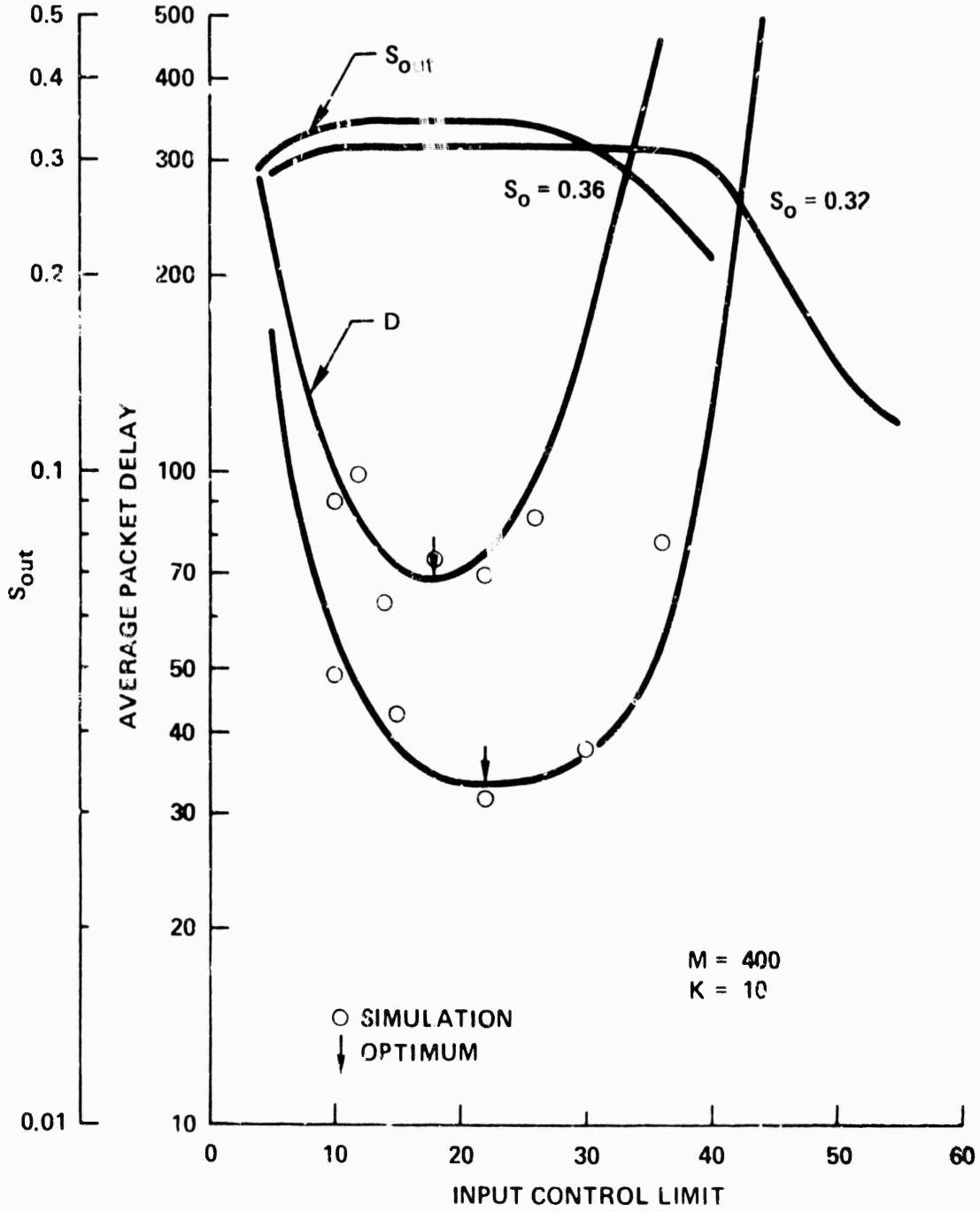


Figure 6-14. Channel Performance Versus ICP Control Limit for M = 400.

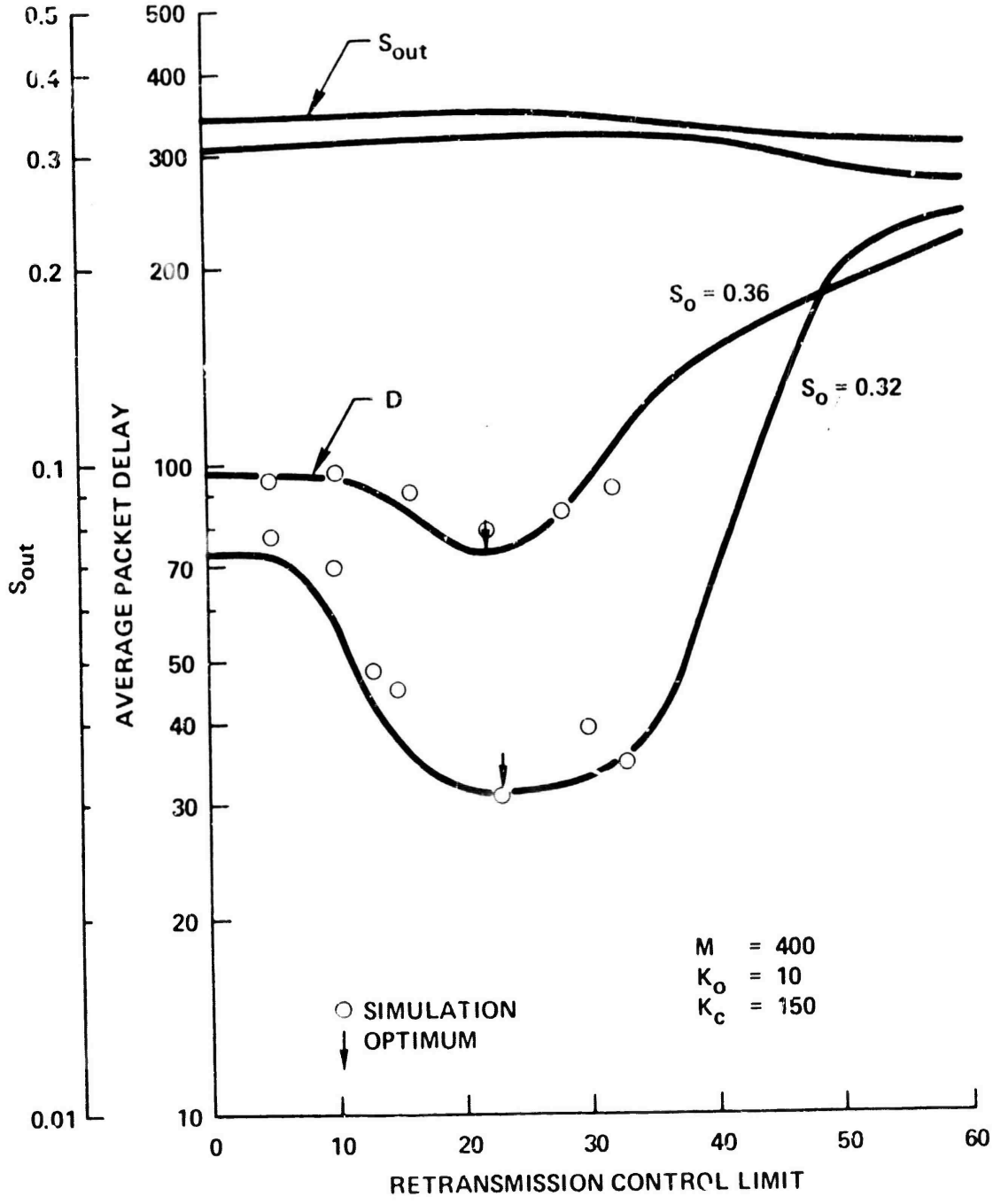


Figure 6-15. Channel Performance Versus RCP Control Limit for M = 400.

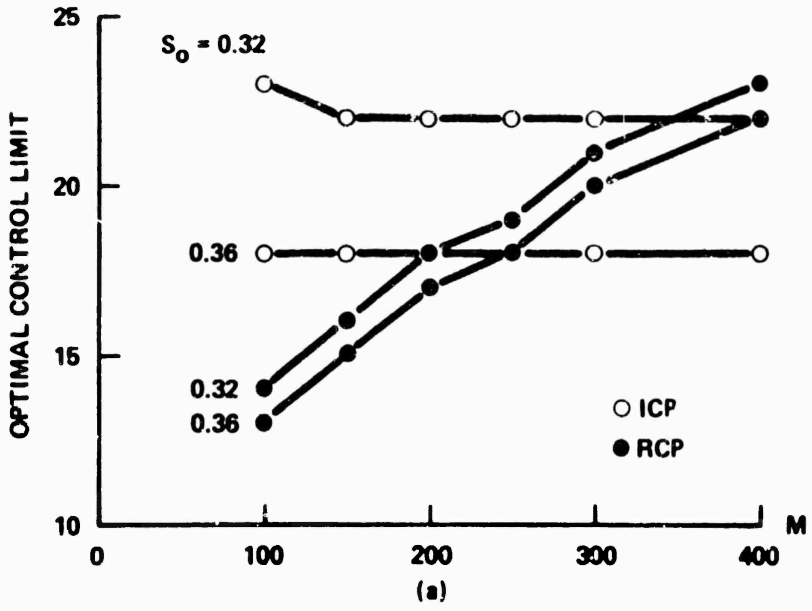
has relatively little effect on the optimal ICP control limit as shown in Fig. 6-16(a). Thus, even if M fluctuates in time in a real system, the same ICP control limit policy is still optimal. Of course, the optimum channel performance must deteriorate as M increases as shown in Fig. 6-16(b). We see also in Fig. 6-16(a) that in the case of RCP, as M (and hence, K_c)* increases, the optimal RCP control limit increases. In Fig. 6-16(b), the optimum D given by ICP and RCP are compared. RCP is found to be slightly better than ICP. However, as M becomes large, K_c must also be large, in which case the trend indicates that ICP is superior to RCP.

We mentioned earlier that for a value of M larger than 500, we run into difficulties with round-off errors such that using POLITE, the optimal control policy can be found only when it is close to the initial control policy. We see here that for a very large M, ICP is superior to RCP. The ICP optimal control limit is also insensitive to M and thus, the same control limit may be used even when M becomes very large.

In Figs. 6-12 to 6-15, we have also indicated simulation results for throughput and delay. Throughput results are shown in Fig. 6-12 only and omitted in the other three figures (but they agree as well with the analytic results as shown in Fig. 6-12). In these simulations, channel control policies are applied assuming that the exact channel backlog size N^t is known to all channel

---

*For both $S_o = 0.32, 0.36$ and corresponding to $M = 100, 150, 200, 250, 300$ and 400, we let $K_c = 20, 40, 60, 80, 100, 150$ respectively.



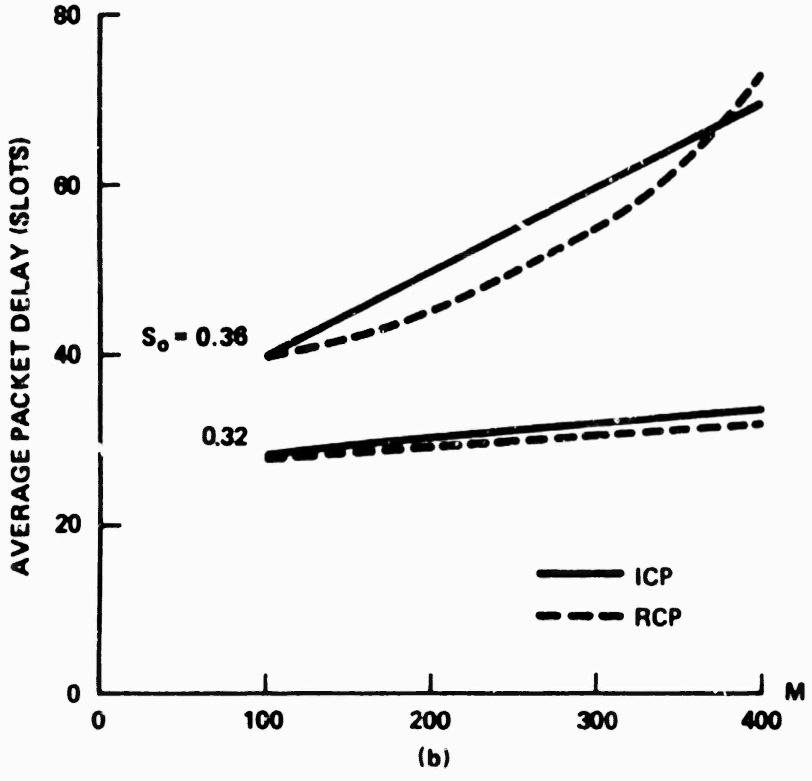


Figure 6-16. ICP and RCP Channel Performance Versus M.

users. However, contrary to the mathematical model, each collided packet is assumed to suffer a fixed delay R and its retransmission to be randomized uniformly over the next K slots. The mathematical model is idealized since R is assumed to be zero while each backlogged packet retransmits in a time slot with probability $p = \frac{1}{R+(K+1)/2}$. (In both cases, the average retransmission delay is the same.) This approximation was examined in Section 5.1 for an uncontrolled channel and found to be very good under the assumption of channel equilibrium. The excellent agreement between the simulation and analytic results presented here demonstrates that this approximation is good even for a dynamically controlled channel. The duration of each simulation run was taken to be 30,000 time slots. The reason for using such a long duration is that in those cases when the control limit $\hat{n}$ is large or when S_o is relatively small, such as 0.32, N^t may exceed $\hat{n}$ only once in a long time. If such time periods are large compared to the duration of a run, the simulation results will not be accurate since we are trying to determine the average value of a random quantity using only a small number of samples.

Optimum throughput-delay tradeoffs

Given a channel control procedure, we consider here the optimum throughput-delay tradeoff corresponding to the boundary of the infeasible region in Fig. 6-6. In Fig. 6-17, given $M = 400$ and a fixed σ, we see that S_{out} is maximized and D minimized by the optimal control limit $n = 22$. With a fixed M, the optimum throughput-delay tradeoff curve is obtained by increasing σ

from zero and for each value of σ, finding the optimal CL and evaluating the optimum channel performance through application of POLITE. Such optimum throughput-delay tradeoffs at fixed values of M are shown in Figs. 6-17 and 6-18 for ICP and RCP respectively. Also shown in these figures is the optimum performance envelope of the infinite population model given in Chapter 3. Note how close the ICP and RCP throughput-delay tradeoff curves are to the optimum envelope. In fact, the $M = 50$ tradeoff curve lies a little below the optimum envelope. This is to be expected since $M = 50$ actually gives rise to a stable channel, in which case the channel performance at the operating point is achieved. Note that these two curves are obtained from two different analytic models based upon different approximations, namely, the first order approximation model in Chapter 3 and the linear feedback model in Chapter 5. It is comforting to see that the two different approximations lead to such close results.

In Figs. 6-19 and 6-20, we show optimum throughput-delay tradeoffs at fixed values of σ for ICP and RCP respectively. ($\frac{1}{\sigma}$ is the average think time of a channel user.) In this case, increasing S_{out} corresponds to increasing M, that is, admitting more channel users. We see that the channel performance improves as the packet generation probability σ increases, since this implies that for the same S_{out}, the number of channel users M is smaller. We considered average think times of 10-30 seconds (see Section 5.1.2). User populations with smaller average think times will probably give

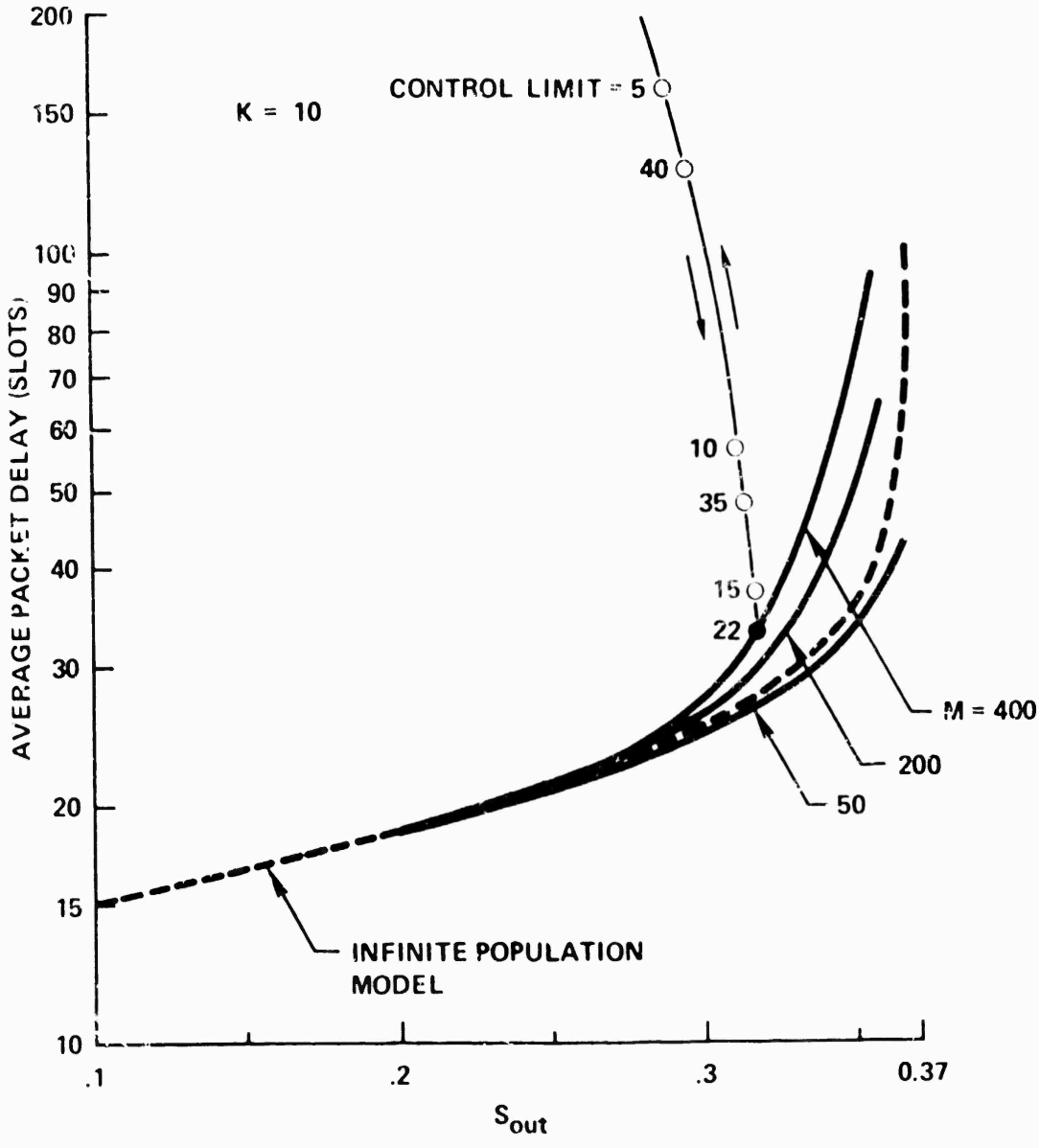


Figure 6-17. ICP Optimum Throughput-Delay Tradeoffs at Fixed M.

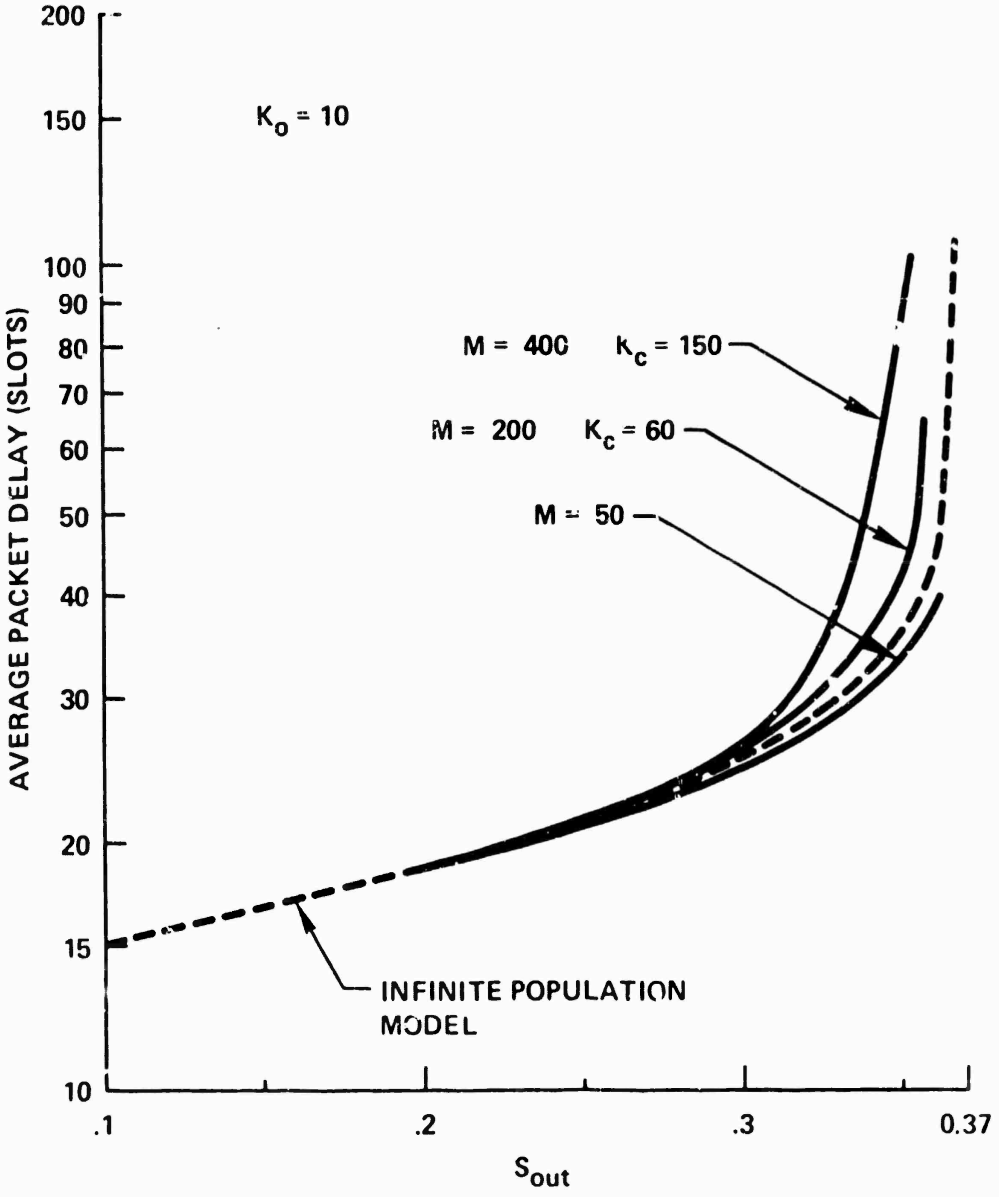


Figure 6-18. RCP Optimum Throughput-Delay Tradeoffs at Fixed M.

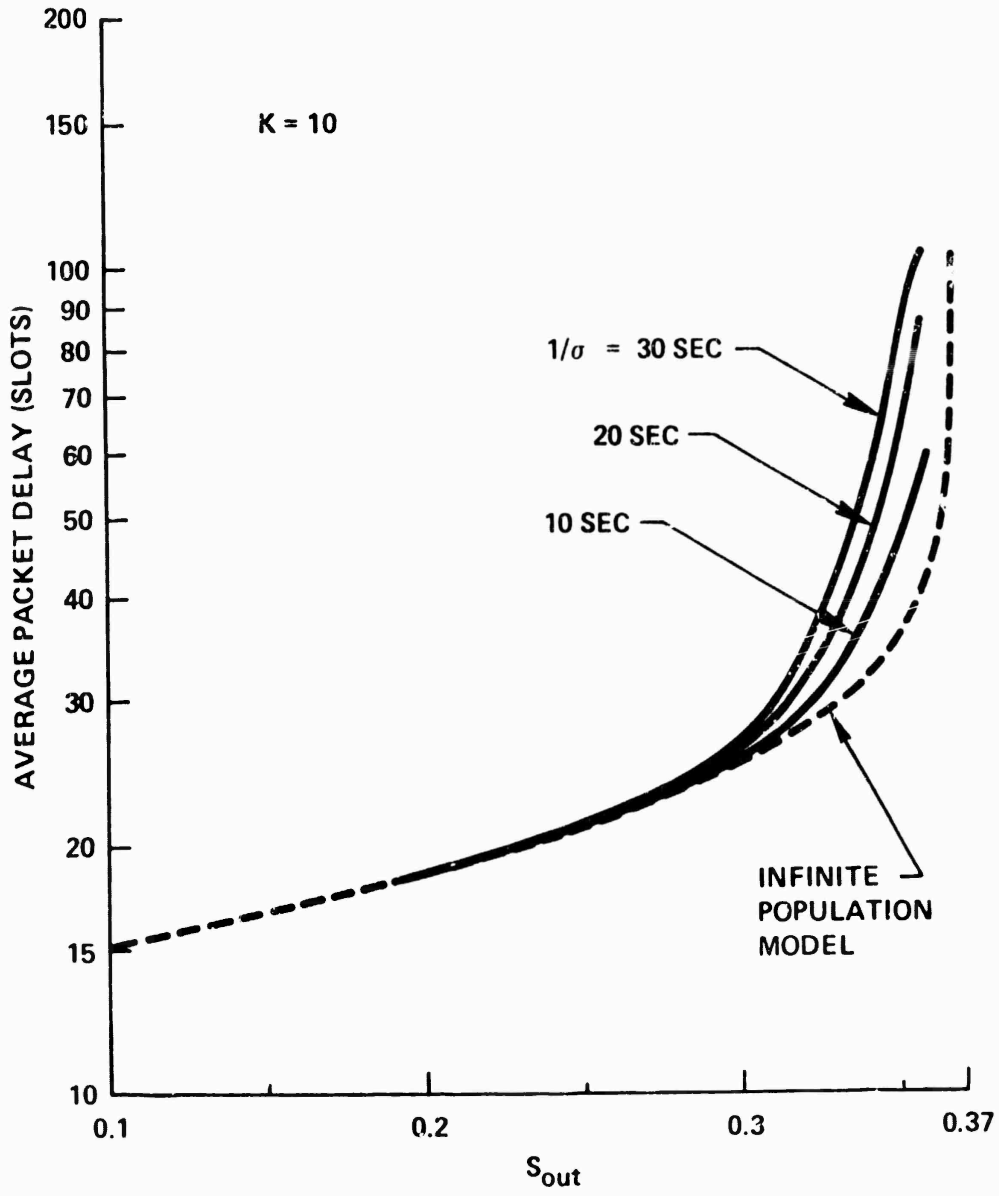


Figure 6-19. ICP Optimum Throughput-Delay Tradeoffs at Fixed σ.

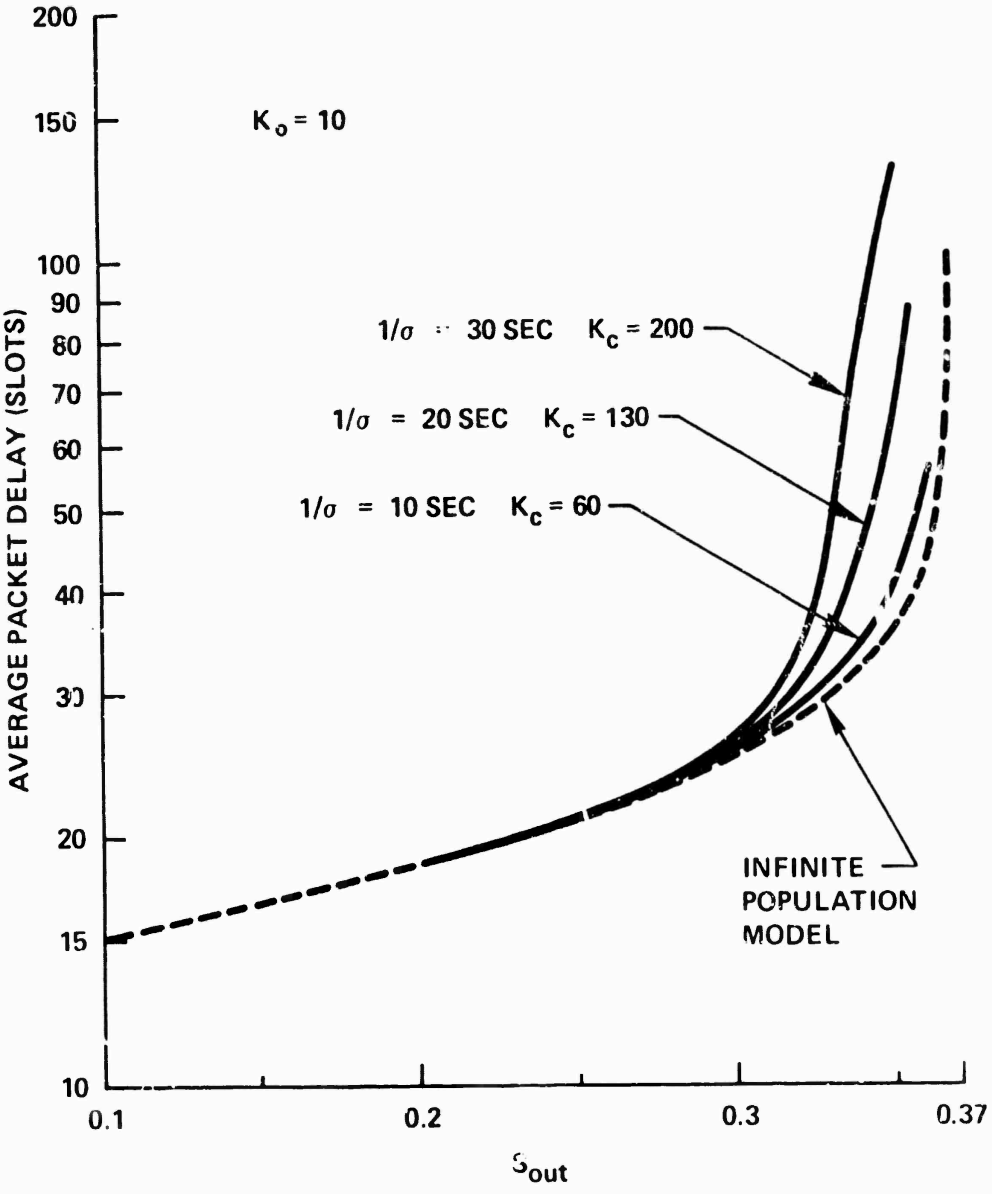


Figure 6-20. RCP Optimum Throughput-Delay Tradeoffs at Fixed σ.

rise to a stable channel, since M must be smaller in this case. Tradeoff curves for larger average think times can be generated by the algorithm POLITE if necessary.

Comparing ICP and RCP in the last four figures, we see that they give rise to almost the same throughput-delay tradeoffs. RCP is slightly better than ICP except when M or $\frac{1}{\sigma}$ is large (e.g., $M > 400$ or $\frac{1}{\sigma} = 30$ seconds).

IRCP channel performance

Recall that the ICP and RCP action spaces are both subspaces of the IRCP action space. Therefore, the channel performance given by IRCP must be better or at least as good as that given by ICP or RCP. This has been verified in all cases we considered. However, in each case, the differences in S_{out} and D among these three channel control procedures are small as shown in Table 6.2 for the four cases involving $M = 200, 400$ and $(n_o, s_o) = (4, 0.32), (7, 0.36)$. Observe that in every instance, IRCP gives the best performance, but only by a very slim margin. Note also that the optimal policy for IRCP is of the form

$$f(i) = \begin{cases} ao & 0 \le i \le \hat{n}_1 \\ ac & \hat{n}_1 < i \le \hat{n}_2 \\ rc & \hat{n}_2 < i \end{cases} \qquad (6.55)$$

which is uniquely specified by $(\hat{n}_1, \hat{n}_2)$. This is similar to a "concatenation" of RCP and ICP control limits! In fact, $\hat{n}_1$ is either equal or very close to the optimal RCP control limit in each case and

		M = 200 S_o = 0.32 (K_c = 60)	M = 200 S_o = 0.36 (K_c = 60)	M = 400 S_o = 0.32 (K_c = 150)	M = 400 S_o = 0.36 (K_c = 150)
$\hat{n}$	ICP	22	18	22	18
$\hat{n}$	RCP	18	17	23	22
$(\hat{n}_1, \hat{n}_2)$	IRCP	(18, 56)	(17, 43)	(23, 116)	(23, 91)
S_{out}	ICP	0.31778	0.34925	0.31807	0.34846
	RCP	0.31817	0.35217	0.31844	0.34715
	IRCP	0.31817	0.35219	0.31844	0.34847
D	ICP	29.857	49.532	33.096	69.237
	RCP	29.085	44.802	31.608	73.588
	IRCP	29.085	44.772	31.608	69.215

Table 6.2 Comparison of ICP, RCP and IRCP.

the use of $\hat{n}_2$ brings about only minor improvement in the channel performance except in the case of $M = 400$ and $S_o = 0.36$. We shall also refer to $\hat{n}_1$ and $\hat{n}_2$ as control limits.

6.7 Practical Control Schemes

The optimal throughput-delay channel performance given in the last section is achievable over an infinite time horizon if the channel users have exact knowledge of the channel state at any time. In a practical system, the channel users often have no means of communication

among themselves other than the multi-access broadcast channel itself. Each channel user must individually estimate the channel state by observing the outcome in each channel slot. Moreover, whatever channel state information available to the channel users is at least one round-trip propagation delay old and may introduce additional errors in the users' estimates if R is large (such as in a satellite channel). Thus, the control action applied based upon an estimate of the channel state may not necessarily be the optimal one at that time, which then will lead to some degradation in channel performance.

Below we first give a procedure for estimating the channel state assuming that the history (i.e., empty slots, successful transmissions or collisions) of the channel is available to all channel users. The optimal ICP, RCP and IRCP control policies will be applied based upon the above estimate. A heuristic control procedure is then proposed which circumvents the state estimation problem. These control procedures are examined through simulations and compared with the optimal throughput-delay results in the previous section. The ability of these control procedures to handle time-varying inputs (with pulses) is also examined. Two other control procedures will then be discussed and some channel design considerations given.

6.7.1 Channel Control-Estimation Algorithms (CONTEST)

Our heuristic procedure for estimating the channel state is based upon the observation that the channel traffic in a time slot is approximately Poisson distributed (see Chapter 4 and Appendix A). Below we present algorithms which implement channel control procedures

studied in the previous sections using the estimated channel state. These channel CONTrol-ESTimation algorithms will be referred to as CONTEST algorithms.

CONTEST algorithms

We give here a procedure for implementing RCP. As before, we let K_o be the operating value and K_c be the control value of K. Suppose $\hat{n}$ is the RCP control limit such that the channel users switch their retransmission K value from K_o to K_c when the channel backlog size exceeds $\hat{n}$ and from K_c to K_o as soon as the channel backlog size drops below $\hat{n}$. We let

$$\hat{G}_o = \hat{n}\, p_o + (M - \hat{n})\sigma \tag{6.56}$$

where from Eq. (5.3)

$$p_o = \frac{1}{R + (K_o + 1)/2}$$

We also define

$$\hat{G}_c = \hat{n}\, p_c + (M - \hat{n})\sigma \tag{6.57}$$

where

$$p_c = \frac{1}{R + (K_c + 1)/2}$$

$\hat{G}_o$ and $\hat{G}_c$ are thus the average channel traffic rates given that the channel backlog size is $\hat{n}$ packets with K equal to K_o and K_c

respectively. Assuming that the channel traffic is approximately Poisson distributed, we define the following critical values (corresponding to the probability of zero channel traffic in a time slot),

$$\hat{f}_o = e^{-\hat{G}_o} \tag{6.58}$$

and

$$\hat{f}_c = e^{-\hat{G}_c} \tag{6.59}$$

Since $K_c > K_o$ we must have

$$\hat{f}_o < \hat{f}_c$$

Suppose each channel user keeps track of the channel history (one round-trip propagation delay ago) within a window frame of W slots as shown in Fig. 6-21. Let $\bar{f}^t$ be the fraction of empty slots in the W slots within the history window for the t^{th} time slot. $\bar{f}^t$ will closely approximate the probability of zero channel traffic

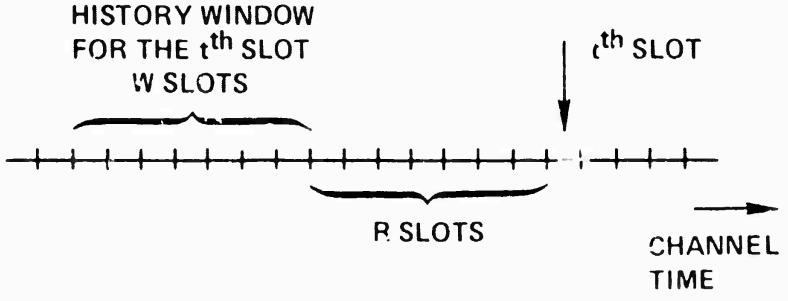


Figure 6-21. The Channel History Window at Time t.

in the t^{th} time slot provided that the channel traffic probability distribution does not change appreciably in $(W + R)$ time slots and the Poisson traffic assumption holds. We give the following CONTEST algorithm to be adopted by each channel user. Let d^t be the control decision at time t.

Algorithm 6.7 (RCP-CONTEST)

This algorithm generates the decision $d^t = K_o, K_c$ at each time point based upon the channel state estimate $\bar{f}^t$ and the RCP control limit $\hat{n}$. Start at step (1) or step (4).

(1) $t \leftarrow t + 1$

$d^t = K_o$

(2) If $\bar{f}^t < \hat{f}_o$, go to (4)

(3) Go to (1)

(4) $t \leftarrow t + 1$

$d^t = K_c$

(5) If $\bar{f}^t > \hat{f}_c$, go to (1)

(6) Go to (4)

Next we consider a similar implementation for ICP. In ICP, the control actions are {accept, reject}. Suppose $\hat{n}$ is the ICP control limit such that the channel always rejects new packet arrivals when the current backlog size exceeds $\hat{n}$ and always accepts new packets when the current backlog size is less than or equal to $\hat{n}$. We let

$$\hat{G}_a = \hat{n}\, p + (M - \hat{n})\sigma \tag{6.60}$$

and

$$\hat{G}_r = \hat{n}\, p \tag{6.61}$$

where

$$p = \frac{1}{R + (K + 1)/2}$$

$\hat{G}_a$ and $\hat{G}_r$ are the average channel traffic rates given that the channel backlog size is $\hat{n}$ packets with the current decision = accept, reject respectively. Again assuming a Poisson channel traffic, we define the following critical values (corresponding to the probability of zero channel traffic in a time slot),

$$\hat{f}_a = e^{-\hat{G}_a} \tag{6.62}$$

$$\hat{f}_r = e^{-\hat{G}_r} \tag{6.63}$$

Since $\hat{G}_a > \hat{G}_r$, we must have

$$\hat{f}_a < \hat{f}_r$$

Algorithm 6.8 (ICP-CONTEST)

This algorithm generates the decision d^t = accept, reject at time t, based upon the channel state estimate $\overline{f}^t$ and ICP control limit $\hat{n}$. Start at step (1) or step (4).

(1) $t \leftarrow t + 1$

d^t = accept

(2) If $\bar{f}^t < \hat{f}_a$ go to (4)

(3) Go to (1)

(4) $t \leftarrow t + 1$

d^t = reject

(5) If $\bar{f}^t > \hat{f}_r$ go to (1)

(6) Go to (4)

To implement IRCP, we assume that the control policy is of the form given in Eq. (6.55) such that it is uniquely specified by the control limits $\hat{n}_1$ and $\hat{n}_2$. To be consistent with this assumption, we shall distinguish only three decision states: ao, ac and rc. We define $\hat{f}_o$ and $\hat{f}_c$ by using $\hat{n}_1$ in Eqs. (6.56)-(6.59), $\hat{f}_{ac}$ and $\hat{f}_{rc}$ by using $\hat{n}_2$ and p_c in Eqs. (6.60)-(6.63), and $\hat{f}_{ao}$ by using $\hat{n}_2$ and p_o in Eqs. (6.60) and (6.62). Since $p_o > p_c > \sigma$ and $\hat{n}_2 > \hat{n}_1$, we have $\hat{f}_{ao} < \hat{f}_o$ and $\hat{f}_{ac} < \hat{f}_c$.

Algorithm 6.9 (IRCP-CONTEST)

This algorithm generates the decision d^t = ao, ac, rc at time t based upon the channel state estimate $\bar{f}^t$ and IRCP control policy $(\hat{n}_1, \hat{n}_2)$. Start at step (1), (4), or (7).

(1) $t \leftarrow t + 1$

d^t = ao

(2) If $\bar{f}^t < \hat{f}_{ao}$ go to (7)

otherwise, if $\bar{f}^t < \hat{f}_o$ go to (4)

(3) go to (1)

(4) $t \leftarrow t + 1$

d^t = ac

(5) If $\bar{f}^t > \hat{f}_c$ go to (1)

otherwise, if $\bar{f}^t < \hat{f}_{ac}$ go to (7)

(6) go to (4)

(7) $t \leftarrow t + 1$

$d^t = rc$

(8) If $\bar{f}^t > \hat{f}_{rc}$ go to (4)

(9) go to (7)

The channel history window

The size W of the channel history window kept by each channel user is very important for successful channel state estimation. If W is too large, we may lose information on the dynamic behavior of the channel such that the necessary actions are taken belatedly. If W is too small, we may get large errors in approximating the probability of zero channel traffic by the fraction of empty slots in the history window. A good initial estimate is that W should be bigger than R and of the same order of magnitude. Below we compare simulation results on channel performance for different values of W.

To implement the channel state estimation procedure, each channel user needs to maintain the channel history for W slots. Since it is only necessary to record whether or not a slot is empty, W bits of information suffice. A possible implementation is depicted schematically in Fig. 6-22. The bit string stored in the shift register represents the channel history in a window of W slots. An empty channel slot is represented by '1' while a nonempty

channel slot is represented by '0'. In the figure, the circle represents a summer, the triangle an attenuator and the square a unit delay of one slot.

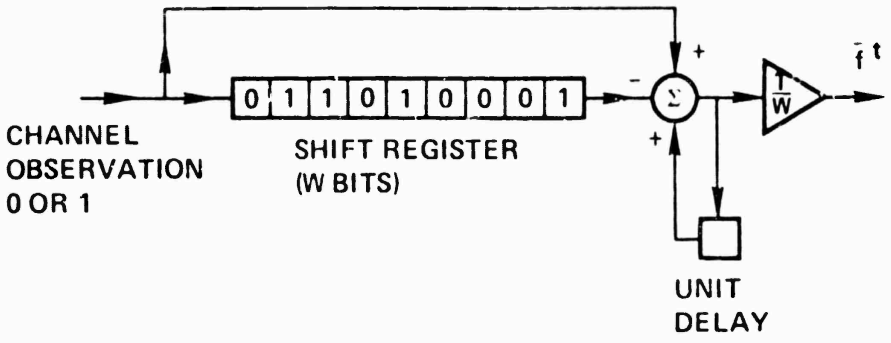


Figure 6-22. Determination of $\bar{f}^t$

Simulation results on the channel performance given by the CONTEST algorithms will be examined below in Section 6.7.3.

6.7.2 Another Retransmission Control Procedure

We describe in this section a simple heuristic control procedure which has the property that when the channel traffic increases the retransmission delays of backlogged packets will also increase. Hence, it will be referred to as the heuristic retransmission control procedure (Heuristic RCP). The advantage of such a control procedure is that it is simple and can be implemented easily without any need for monitoring the channel history and estimating the channel state. In the next section, this and the above CONTEST algorithms will be compared through simulations.

The Control Scheme

For a backlogged packet with m previous channel collisions, the uniform retransmission randomization* interval is taken to be $K = K_m$ where K_m is a monotone nondecreasing function in m.

When the channel traffic increases, the probability of channel collision increases. As a result, the "effective" value of K increases. If K_m is a steep enough function in m, we see that channel saturation will be prevented. An effective value of K can be defined only with respect to a specific performance measure (e.g., average packet delay). To illustrate the effect of the function K_m, we derive below the average value of K as a function of q (the probability of successful transmission). Let

$$r_i = \text{Prob[a packet retransmits } i \text{ times before success]} = (1 - q)^i q \qquad i \geq 1$$

Case 1 $\quad K_m = K_2$ for $m \geq 2$ and $K_2 > K_1$

$$\bar{K} = \text{average value of } K = \frac{1}{1 - q} \sum_{i=1}^{\infty} r_i \sum_{m=1}^{i} \frac{K_m}{i}$$

*Note that the same control scheme can be extended to geometric retransmission randomization by letting $p = p_m$ where p_m is a monotone nonincreasing function.

$$= \frac{1}{1-q} \sum_{i=1}^{\infty} (1-q)^i \, q \left(\frac{K_1}{i} + \frac{i-1}{i} K_2 \right)$$

$$= K_2 + \frac{q \, \ell n \, q}{1-q} (K_2 - K_1) \qquad (6.64)$$

which is equal to K_1 at $q = 1$ and increases to K_2 as q decreases to zero; ℓn is the natural logarithm function.

Case 2 $\quad K_m = K_3$ for $m \geq 3$ and $K_3 > K_2 > K_1$

$$\bar{K} = \frac{1}{1-q} \sum_{i=1}^{\infty} r_i \sum_{m=1}^{i} \frac{K_m}{i}$$

$$= \frac{1}{1-q} \sum_{i=2}^{\infty} (1-q)^i \, q \left(\frac{K_1}{i} + \frac{K_2}{i} + \frac{i-2}{i} K_3 \right)$$

$$+ (1-q) \, q \, K_1$$

$$= K_3 + (K_3 - K_2) \, q + \frac{q \, \ell n \, q}{1-q} (2K_3 - K_1 - K_2) \qquad (6.65)$$

which is equal to K_1 at $q = 1$ and increases to K_3 as q decreases to zero.

Case 3 $\quad K_m = m K \qquad m \geq 1$

$$\bar{K} = \frac{1}{1-q} \sum_{i=1}^{\infty} r_i \sum_{m=1}^{i} \frac{K_m}{i}$$

$$= \frac{K}{1 - q} \sum_{i=1}^{\infty} \frac{(1 - q)^i q}{i} \sum_{m=1}^{i} m$$

$$= \frac{K}{2} (1 + \frac{1}{q}) \qquad (6.66)$$

which is equal to K at $q = 1$ and increases to infinity as q decreases to zero.

The above results indicate that the average value of K behaves in the desired manner, namely, $\bar{K}$ increases as q decreases due to an increasing channel traffic. This behavior is similar to that of the retransmission control procedure. That is why the above procedure is called Heuristic RCP. Below we examine the CONTEST algorithms and Heuristic RCP through simulations.

6.7.3 Simulation Results

We summarize in Tables 6.3-6.6, throughput-delay results for the following channel load lines,

(1) $M = 200$, $(n_o, S_o) = (4, 0.32)$

(2) $M = 400$, $(n_o, S_o) = (4, 0.32)$

(3) $M = 200$, $(n_o, S_o) = (7, 0.36)$

(4) $M = 400$, $(n_o, S_o) = (7, 0.36)$

In all cases, K_o is equal to 10. K_c is taken to be 60 and 150 for M equal to 200 and 400 respectively. Included in these tables are (a) optimum POLITE results for ICP, RCP and IRCP, (b) simulation results for ICP and RCP using optimal control policies and under the assumption of perfect channel state information, (c) simulation results

for the CONTEST algorithms using ICP and RCP optimal control policies, and (d) simulation results for Heuristic RCP. Each simulation run is identified by the seed supplied to the random number generator. The duration of each simulation run was taken to be 30,000 time slots. IRCP was not tested by simulation since the optimal value of $\hat{n}_2$ is in all cases so large that within the simulation duration, the channel state N^t (almost surely) will not exceed it; the control procedure becomes effectively RCP specified by $\hat{n}_1$.

The ICP-CONTEST algorithm was tested with channel history window sizes of 20, 40, 60 and 80 time slots. We see from Tables 6.3-6.6 that $W = 40$ appears to give the best throughput-delay results. Note that for $R = 12$ and $K = K_o = 10$, $W = 40$ is approximately twice $R + K$.

The RCP-CONTEST algorithm was also tested with various values of W . In this case, K takes on two values, K_o and K_c where $K_c = 60$ or 150 depending on M . There is no clear-cut optimal W . It appears that $W = 60$ is a good choice for $K_c = 60$ and $M = 200$ while $W = 80$ is a good choice for $K_c = 150$ and $M = 400$.

Results for $S_o = 0.32$ and $M = 200, 400$ are shown in Tables 6.3 and 6.4. We see that there is no significant degradation in channel performance (from the optimum) given by the CONTEST algorithms and Heuristic RCP. The CONTEST algorithms, however, seem to have an edge over Heuristic RCP. The excellent performance of the CONTEST algorithms can be attributed to the flatness of S_{out} and D near the optimum as a function of the control limit (see Fig.

CONTROL SCHEME	RANDOM NUMBER GENERATOR SEED IN SIMULATION	S_{out}	D
ICP	-----	0.31778	29.857
RCP	-----	0.31817	29.085
IRCP	-----	0.31817	29.085
ICP	39474	0.315	33.427
RCP	78453	0.318	28.824
ICP-CONTEST W = 20	73645	0.314	40.893
" " W = 40	39587	0.315	30.514
" " W = 60	59478	0.317	32.355
" " W = 80	54857	0.318	35.809
RCP-CONTEST W = 20	49784	0.315	33.052
" " W = 40	58474	0.322	33.335
" " W = 60	20494	0.319	32.138
" " W = 80	10398	0.317	32.501
Heuristic RCP $\begin{cases} K_1 = 10 \\ K_m = 60 \end{cases}$ $m \geq 2$	18867	0.316	33.720
	61111	0.315	34.554
Heuristic RCP $\begin{cases} K_1 = 10 \\ K_2 = 60 \\ K_m = 120 \end{cases}$ $m \geq 3$	63037	0.310	35.425
	07275	0.316	34.635

Table 6.3 Throughput-delay results of a controlled channel ($M = 200$, $S_o = 0.32$)

CONTROL SCHEME	RANDOM NUMBER GENERATOR SEED IN SIMULATION	S_{out}	D
ICP	-----	0.31807	33.096
RCP	-----	0.31844	31.608
IRCP	-----	0.31844	31.608
ICP	84023	0.315	31.427
RCP	40393	0.317	31.023
ICP-CONTEST W = 20	94875	0.315	43.262
" " W = 40	39848	0.314	34.723
" " W = 60	74945	0.312	53.240
" " W = 80	94875	0.316	39.112
RCP-CONTEST W = 20	49784	0.313	41.087
" " W = 40	58474	0.319	43.379
" " W = 60	20494	0.318	38.821
" " W = 80	10398	0.317	40.068
" " W = 100	64945	0.314	35.689
" " W = 120	18494	0.319	47.149
Heuristic RCP $K_1 = 10$ $m \geq 2$	57298	0.316	45.150
Heuristic RCP $K_m = 150$ $m \geq 2$	16489	0.316	44.750
Heuristic RCP $K_1 = 10$, $K_2 = 100$, $K_m = 200$ $m \geq 3$	38687	0.312	42.040
Heuristic RCP $K_1 = 10$, $K_2 = 100$, $K_m = 200$ $m \geq 3$	46534	0.311	43.136

Table 6.4 Throughput-delay results of a controlled channel ($M = 400$, $S_o = 0.32$)

CONTROL SCHEME	RANDOM NUMBER GENERATOR SEED IN SIMULATION	S_{out}	D
ICP	-----	0.34925	49.552
RCP	-----	0.35217	44.802
IRCP	-----	0.35219	44.772
ICP	18654	0.346	59.111
RCP	95646	0.348	48.655
ICP-CONTEST W = 20	18947	0.331	83.664
" " W = 40	53857	0.339	77.357
" " W = 60	89574	0.330	87.614
" " W = 80	10394	0.332	73.310
RCP-CONTEST W = 20	03847	0.347	67.900
" " W = 40	39475	0.345	50.853
" " W = 60	60389	0.345	50.534
" " W = 80	10489	0.347	51.787
Heuristic RCP $K_1 = 10$, $K_m = 60$ $m \geq 2$	94854	0.349	48.535
	37776	0.344	46.116
Heuristic RCP $K_1 = 10$, $K_2 = 60$, $K_m = 120$ $m \geq 3$	94854	0.350	50.267
	18495	0.347	54.583

Table 6.5 Throughput-delay results of a controlled channel ($M = 200$, $S_o = 0.36$)

CONTROL SCHEME	RANDOM NUMBER GENERATOR SEED IN SIMULATION	S_{out}	D
ICP	-----	0.34846	69.237
RCP	-----	0.34715	73.588
IRCP	-----	0.34847	69.215
ICP	28879	0.343	73.524
RCP	11217	0.350	79.270
ICP-CONTEST W = 20	38457	0.334	128.460
" " W = 40	06348	0.330	98.994
" " W = 60	74948	0.336	126.143
" " W = 80	74394	0.332	119.628
RCP-CONTEST W = 20	38457	0.341	99.701
" " W = 40	06348	0.335	97.676
" " W = 60	74948	0.343	97.048
" " W = 80	74394	0.340	91.833
" " W = 100	38373	0.343	107.722
" " W = 120	93875	0.337	99.192
Heuristic RCP $K_1 = 10$, $K_m = 150$, $m \geq 2$	99581	0.344	66.327
	54857	0.352	70.590
Heuristic RCP $K_1 = 10$, $K_2 = 100$, $K_m = 200$, $m \geq 3$	38378	0.345	81.324
	36949	0.348	62.662

Table 6.6 Throughput-delay results of a controlled channel ($M = 400$, $S_0 = 0.36$)

6-12 to 6-15). This property is typical of channel load lines specified by small to moderate values of S_o. We see in the same figures that the flatness in S_{out} and D near the optimum is not as pronounced when $S_o = 0.36$. This explains the more significant degradation in channel performance given by the CONTEST algorithms shown in Tables 6.5-6.6. Note that for $S_o = 0.36$, Heuristic RCP gives much better throughput-delay results than the CONTEST algorithms.

In Figs. 4-2 and 4-4, it was shown that in an uncontrolled channel, a channel input rate of 0.8 packet/slot sustained for 100 time slots was enough to cripple the channel. In Figs. 6-23 and 6-24, we show by simulations that under similar but more severe circumstances both the IRCP-CONTEST algorithm and Heuristic RCP prevented the channel from going into saturation. In these simulations, the normal channel load line was given by $M = 400$ and $(n_o, S_o) = (4, 0.32)$ both before and after the pulse. During a period of 200 slots (namely, the time period 1000-1200), the packet generation probability σ was increased such that $M\sigma = 1.0$ packet/slot. Observe that both algorithms handled the sudden influx of new packets with ease. In both cases, the channel throughput, instead of vanishing to zero as in an uncontrolled channel, maintained at a high rate and within less than 3000 slots, the channel returned to almost normal operation.

6.7.4 Other Proposed Schemes

Other channel control procedures have been proposed by Metcalfe [METC 73A] and Rettberg [RETT 73C].

In Metcalfe's proposal, Q is defined to be the current number of channel users who have a packet ready to transmit over the channel.

INPUT PARAMETERS:

NUMBER OF TERMINALS M = 400 , PROPAGATION DELAY R = 12
FOR THE TIME PERIOD 1-1000, INPUT RATE Mσ = 0.3232
FOR THE TIME PERIOD 1001-1200, INPUT RATE M 1.0
FOR THE TIME PERIOD 1201-6000, INPUT RATE Mσ = 0.3232
RETRANSMISSION CONTROL LIMIT = 23, INPUT CONTROL LIMIT = 116
K_o = 10 , K_c = 150 , WINDOW SIZE W = 60
RANDOM NUMBER GENERATOR SEED = 81724

AVERAGE VALUES IN 200 TIME SLOT PERIODS:

TIME PERIOD	THROUGHPUT RATE- S	TRAFFIC RATE- G	PACKET DELAY- D	FRACTION EMPTY	AVERAGE BACKLOG	PACKETS REJECTED
1 - 200	0.290	0.625	30.29	0.555	5.5	0
201 - 400	0.325	0.700	34.06	0.505	6.9	1
401 - 600	0.235	0.450	23.67	0.635	2.9	0
601 - 800	0.295	0.625	31.76	0.565	5.9	0
801 - 1000	0.325	0.850	42.57	0.455	9.3	0
1001 - 1200	0.205	2.345	524.32	0.190	52.0	49
1201 - 1400	0.345	1.330	389.65	0.325	75.2	13
1401 - 1600	0.355	0.880	188.11	0.435	51.5	0
1601 - 1800	0.375	0.735	179.37	0.460	34.9	0
1801 - 2000	0.225	1.295	297.85	0.385	33.7	5
2001 - 2200	0.325	1.005	530.77	0.415	55.3	21
2201 - 2400	0.380	0.905	127.32	0.365	16.7	0
2401 - 2600	0.355	0.485	27.64	0.605	3.1	0
2601 - 2800	0.290	0.430	20.86	0.640	2.3	0
2801 - 3000	0.345	0.745	35.38	0.485	7.2	0
3001 - 3200	0.300	0.455	17.62	0.635	3.4	0
3201 - 3400	0.280	0.615	28.48	0.575	5.6	0
3401 - 3600	0.390	0.810	37.90	0.425	7.9	0
3601 - 3800	0.330	0.655	30.65	0.520	5.6	0
3801 - 4000	0.300	0.390	19.30	0.655	1.7	0
4001 - 4200	0.315	0.615	29.24	0.560	5.1	0
4201 - 4400	0.335	0.600	24.51	0.545	4.5	0
4401 - 4600	0.300	0.450	24.32	0.630	2.6	0
4601 - 4800	0.280	0.480	25.29	0.625	3.7	0
4801 - 5000	0.285	0.585	32.07	0.580	5.3	0
5001 - 5200	0.330	0.570	26.41	0.555	4.3	0
5201 - 5400	0.335	0.550	23.67	0.560	3.7	0
5401 - 5600	0.335	0.640	28.81	0.530	5.2	0
5601 - 5800	0.275	0.410	21.56	0.660	2.4	0
5801 - 6000	0.285	0.445	22.35	0.645	2.7	0

Fig. 6-23 Simulation run for IRCP-CONTEST subject to a channel input pulse.

INPUT PARAMETERS:

NUMBER OF TERMINALS M = 400 , PROPAGATION DELAY R = 12

FOR THE TIME PERIOD 1-1000, INPUT RATE $M\sigma$ = 0.3232

FOR THE TIME PERIOD 1001-1200, INPUT RATE $M\sigma$ = 1.0

FOR THE TIME PERIOD 1201-6000, INPUT RATE $M\sigma$ = 0.3232

K_1 = 10 K_m = 150 ($m \geq 2$)

RANDOM NUMBER GENERATOR SEED = 67289

AVERAGE VALUES IN 200 TIME SLOT PERIODS:

TIME PERIOD	THROUGHPUT RATE- S	TRAFFIC RATE- G	PACKET DELAY- D	FRACTION EMPTY	AVERAGE BACKLOG
1 - 200	0.285	0.395	19.877	0.665	2.1
201 - 400	0.320	0.390	16.328	0.650	1.2
401 - 600	0.255	0.425	22.824	0.660	2.8
601 - 800	0.290	0.475	26.172	0.630	4.0
801 - 1000	0.325	0.570	28.554	0.570	5.7
1001 - 1200	0.230	2.395	34.109	0.120	68.8
1201 - 1400	0.285	1.695	141.333	0.215	112.6
1401 - 1600	0.310	1.500	273.177	0.230	91.8
1601 - 1800	0.375	1.415	288.693	0.190	68.5
1801 - 2000	0.280	1.110	224.661	0.175	53.1
2001 - 2200	0.360	1.240	257.333	0.300	48.8
2201 - 2400	0.355	0.925	193.986	0.395	31.3
2401 - 2600	0.385	0.655	122.818	0.490	15.2
2601 - 2800	0.320	0.565	68.094	0.565	8.8
2801 - 3000	0.280	0.420	39.357	0.660	5.5
3001 - 3200	0.295	0.495	31.678	0.615	6.3
3201 - 3400	0.265	0.630	45.000	0.545	11.7
3401 - 3600	0.350	0.750	37.057	0.485	13.3
3601 - 3800	0.310	0.465	65.274	0.625	8.2
3801 - 4000	0.275	0.520	33.618	0.610	7.7
4001 - 4200	0.330	0.480	34.652	0.595	5.2
4201 - 4400	0.325	0.615	29.585	0.540	7.5
4401 - 4600	0.370	0.525	38.608	0.560	7.6
4601 - 4800	0.260	0.705	44.250	0.550	15.9
4801 - 5000	0.375	0.720	63.520	0.460	11.1
5001 - 5200	0.350	0.615	41.729	0.520	9.2
5201 - 5400	0.285	0.475	29.368	0.625	6.6
5401 - 5600	0.315	0.510	36.460	0.595	4.9
5601 - 5800	0.290	0.425	24.190	0.650	4.1
5801 - 6000	0.305	0.450	28.738	0.610	4.7

Fig. 6-24 Simulation run for Heuristic RCP subject to a channel input pulse.

(Q is different from our channel backlog size N^t since Q includes both backlogged and newly generated packets.) The control scheme suggested is that each of the Q channel users transmits in the next time slot with probability $\frac{1}{Q}$. This strategy maximizes the expected channel throughput in the next time slot (provided that Q is known exactly) and is referred to as throughput maximizing retransmission control. The channel performance given by this control scheme was studied through a steady-state analysis by Metcalfe [METC 73A]. However, the channel performance given by this control scheme in a dynamic environment (either through analysis or simulation) has not been studied.

Rettberg's proposal is concerned with satellite communication involving a small number (e.g., M = 2 to 10) of stations, each of which has buffering and scheduling capabilities. In Rettberg's scheme, newly generated packets attempt transmission over the channel without any delay. Previously collided packets form a queue at each station. Each station has a "gating" probability x of transmitting the packet at the head of its (backlog) queue in a time slot. Rettberg suggested that the gating probability may be chosen such that $Mx + S \leq 1$ where S is the channel input rate of new packets. Since in this case the channel traffic rate G is forced to be less than or equal to one,* no channel saturation will occur. Simulations [RETT 73C] supported this claim.

This scheme may be referred to as probability division multiplexing (PDM). Each channel user, instead of getting a fixed fraction

---

*$G = Mx\rho + S$, where ρ is the probability that a station's backlog queue is nonempty.

of the communication channel capacity such as in time division multiplexing (TDM) or frequency division multiplexing (FDM), now gets a random fraction of the channel capacity through the gating probability x. Thus, similar to TDM and FDM, this scheme will work quite well when M is small and each station has a relatively "smooth" input source. However, when M is large and each user has a bursty input source, PDM will suffer from the same pitfalls of FDM and TDM. That is, many channel users will often have an empty backlog queue (while others have very long queues). As a result, the actual channel traffic rate is very low, which gives rise to a small channel throughput rate. However, the average packet delay is high, since a small x (due to a large M) has been adopted. In this case, some scheme which allocates gating probabilities x_i to channel users dynamically as a function of their instantaneous transmission requirements may prove useful. (The constraint is now $\sum_{i=1}^{M} x_i + S \leq 1$.)

6.7.5 Channel Design Considerations

Consider the design of a slotted ALOHA channel characterized by the linear feedback model. Given M, σ and K, the channel load line and the equilibrium contour may intersect in three different ways depicted in Figs. 5-6 (a), (b) and (d).

In Fig. 5-6(d), the channel is overloaded in the sense that the globally stable equilibrium point corresponds to the channel saturation point. This situation should always be avoided (e.g., by reducing the number of channel users).

In Fig. 5-6(a), the channel operating point (n_o, S_o) is also the globally stable equilibrium point.

In this case, the assumption of channel equilibrium at (n_o, S_o) is valid. Hence, no channel control is necessary.

We have been mostly concerned with the dynamic control of an unstable channel such as shown in Fig. 5-6(b).

Consider the $K = 10$ equilibrium contour in Fig. 5-3. Given an average user think time = $\frac{1}{\sigma}$ (where $-\frac{1}{\sigma}$ is the slope of the channel load line), there is a maximum value of M such that the channel is stable. For example, if $\frac{1}{\sigma} = 615$ slots ($\tilde{=}$ 14 seconds), the maximum number of channel users is approximately 100 without rendering the channel unstable. At this value of M, the channel throughput rate $S_o \tilde{=} 0.162$ and the average packet delay $D \tilde{=} 17.5$ slots (0.394 second). If we want to increase the channel utilization (throughput) by increasing the number of channel users M, one of several things can be done:

(1) Do nothing.

(2) Increase K.

(3) Dynamic channel control.

Suppose M is 150 giving $S_o \tilde{=} 0.244$. The channel is now unstable, but from results in Chapter 5, has a channel FET of several days. If this is an acceptable channel failure rate, no external control is necessary except to restart the channel whenever it goes into saturation.

Increasing K from 10 to 60 allows the channel to support up to 200 users at $S_o \tilde{=} 0.32$. But now the throughput-delay tradeoff curve for $K = 60$ is much above the optimum performance envelope in Fig. 3-4. In Fig. 3-5, we see that for $S_o = 0.32$, $D = 45.5$ slots (1.02 second).

Dynamic channel control can give rise to a stable channel as well as providing a throughput-delay tradeoff close to the optimum envelope. For example, consider the results in Table 6.3 for $M = 200$ and $S_o = 0.32$. Under the assumption of perfect channel state information, a channel throughput-delay tradeoff very close to the optimum envelope is possible as shown in Figs. 6-17 to 6-20 for ICP and RCP. Without perfect channel state information, we have shown by simulations that throughput-delay results close to the optimum envelope can still be achieved using the CONTEST algorithms up to $S_o = 0.32$. (Recall that this is a consequence of the amazing flatness of S_{out} and D near the optimum except when S_o is large.) In any case, the channel operating point probably should not be designed with a value of $S_o > 0.32$. For $S_o > 0.32$, even if it is possible to achieve the optimum envelope, the incremental gain in channel throughput is at the expense of a sizable increase in delay.

In a real system, it is imaginable that the channel input may vary with time (say M fluctuating between say 100 to 200 in the above example). We must emphasize the fact that the control algorithms considered have been designed to control statistical channel fluctuations under the assumption of a stationary channel input.

Although we showed that they can temporarily handle very high channel input rates (see Figs. 6-23 to 6-24), other control mechanisms should be designed into the system to make sure that an overloaded channel such as depicted in Fig. 5-6(d) does not prevail for any long period of time (e.g., by limiting the maximum number of users who can "sign-on" and become active channel users).

We showed earlier that IRCP gives a channel performance at least as good as ICP and RCP. Furthermore, with two control limits $\hat{n}_1$ and $\hat{n}_2$, it acts like RCP (with $\hat{f}_1$) under normal channel conditions, but has a second "defense" in $\hat{n}_2$ whenever the channel traffic increases to a very high value. Comparing IRCP-CONTEST and Heuristic RCP, we see that the latter is easier to implement and exhibited in several simulations better channel performance for a heavily loaded channel ($S_o = 0.36$). However, under a normal load (say $S_o \leq 0.32$), IRCP-CONTEST is superior to Heuristic RCP. This is because Heuristic RCP introduces longer delays to collided packets even when these packets are just unlucky in light channel traffic. On the other hand, in IRCP, control actions are not exerted until the channel traffic exceeds some critical values.

CHAPTER 7

MULTI-PACKET MESSAGE DELAY AND SATELLITE RESERVATION SCHEMES

In a packet switched network, "messages" generated by external sources for transmission over the network are broken into fixed size packets. Up to now we have assumed that all messages generated are fixed length single packets. We have also used the average packet delay as our channel performance measure. This assumption is indeed justified in an interactive computer communications environment. Measurement results [KLEI 74B] indicate that 96 percent of the ARPA network traffic consists of messages shorter than a single packet. However, there are situations in which the average packet delay is not an appropriate channel performance measure: for example, the transfer of long data files and the transmission of digital voice messages [BAYL 73]. In these cases, a more appropriate performance measure may be the average message delay, namely, the delay incurred by a message from the time it is ready for transmission until when all packets in the message have been correctly received at the message destination.

A satellite reservation system has been studied for multi-packet message arrivals by Roberts [ROBE 73]. In this system, the satellite channel is dynamically partitioned into a slotted ALOHA channel for broadcasting reservation requests and a scheduled channel for transmitting multi-packet blocks of data. Since the minimum delay in this case is two satellite round-trip propagation times

($\tilde{=}$ 0.5 sec.), this system is preferable if a significant fraction of the channel input source consists of multi-packet messages and if the average message delay is the relevant channel performance measure.

In this chapter, we first derive an approximate formula for the average message delay in a slotted ALOHA channel. Next, Roberts' reservation system will be introduced. Two other satellite reservation schemes will also be described; these two schemes may be used if there is only a small number of channel users and if the channel input source has constant as well as random components. The reservation schemes, by reducing the amount of channel collisions, are capable of providing channel throughput rates well in excess of the slotted ALOHA channel capacity.

7.1 Multi-Packet Message Delay

In this section, we consider a slotted ALOHA model such as the infinite population model in Chapter 3. However, each arrival is now a message of $\tilde{L}$ packets (where $\tilde{L}$ is an integer-valued random variable specified by some probability distribution). A good approximation for the message delay is the delay incurred by the packet in the message with the most number of retransmissions. Define

p_n = Prob[collision/transmission of a new packet]

p_t = Prob[collision/transmission of a previously collided packet]

Thus, $p_n = 1 - q_n$ and $p_t = 1 - q_t$ where q_n and q_t are specified by S and K in Chapter 3. Let C be a random variable representing

the number of channel collisions a packet incurs before its successful transmission. We then have

$$\text{Prob}[C = i] = \begin{cases} 1 - p_n & i = 0 \\ p_n p_t^{\,i-1}(1 - p_t) & i \geq 1 \end{cases} \tag{7.1}$$

$$\text{Prob}[C \leq i] = 1 - p_n p_t^{\,i} \tag{7.2}$$

We shall assume that all packets in a message have independent indentically distributed numbers of channel collisions. Let C_ℓ be the maximum of ℓ independent random variables with identical distributions given by Eq. (7.2). Hence,

$$\begin{aligned} \text{Prob}[C_\ell \leq i] &= (\text{Prob}[C \leq i])^{\ell} \\ &= (1 - p_n p_t^{\,i})^{\ell} \end{aligned} \tag{7.3}$$

Define the expectation of C_ℓ to be

$$\begin{aligned} E_\ell = E[C_\ell] &= \sum_{i=0}^{\infty} \text{Prob}[C_\ell > i] \\ &= \sum_{i=0}^{\infty} [1 - (1 - p_n p_t^{\,i})^{\ell}] \\ &= \sum_{i=0}^{\infty} [\ell\, p_n p_t^{\,i} - \binom{\ell}{2} p_n^{\,2} p_t^{\,2i} + \binom{\ell}{3} p_n^{\,3} p_t^{\,3i} - \\ &\qquad \cdots\cdots (-1)^{\ell+1} p_n^{\,\ell} p_t^{\,\ell i}] \\ &= \sum_{j=1}^{\ell} (-1)^{j+1} \binom{\ell}{j} \frac{p_n^{\,j}}{1 - p_t^{\,j}} \end{aligned} \tag{7.4}$$

The average delay for a message of ℓ packets is thus approximated by

$$D_\ell = R + \ell + E_\ell[R + (K + 1)/2] \tag{7.5}$$

where $R + (K + 1)/2$ is the average retransmission delay. Note that no buffer scheduling delays are included in this estimate. Thus, the actual average message delay will probably be slightly larger than D_ℓ. Using the above estimate, the average message delay for the channel is given by

$$D_{mess} = \sum_\ell D_\ell \cdot \text{Prob}[\tilde{L} = \ell] \tag{7.6}$$

D_ℓ has been evaluated for $\ell = 1, 2, 4, 8, 20$ and plotted in Fig. 7-1 using numerical values of q_n and q_t for $K = 15$ in the infinite population model in Chapter 3. Thus, the $\ell = 1$ contour in Fig. 7-1 is the same as the $K = 15$ contour in Fig. 3-4. Several simulation points are also shown for $\ell = 4$ and 8. These simulations were performed for the finite population model in Chapter 3 with 20 users and $K = L = 15$. Note that all simulation delay values are larger than their corresponding analytic values since buffer scheduling delays are included in the simulation message delay. Assuming that the channel input is equally divided between single-packet and eight-packet messages, the average message delay for the channel is shown in Fig. 7-2.

Simulations also indicate that when the channel input consists of many multi-packet messages, the slotted ALOHA channel is "more unstable" than before. It may be possible to extend the stability

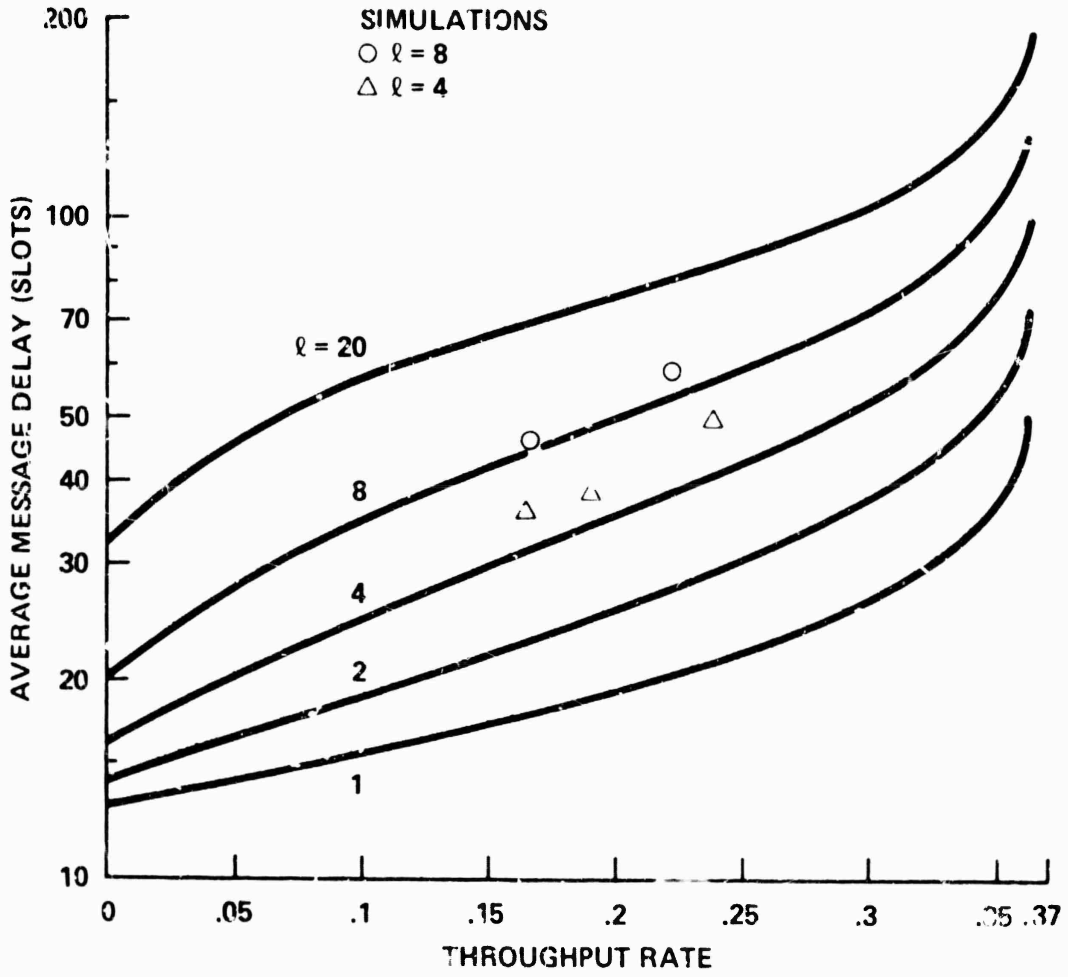


Figure 7-1. Multi-Packet Message Delay Versus Throughput.

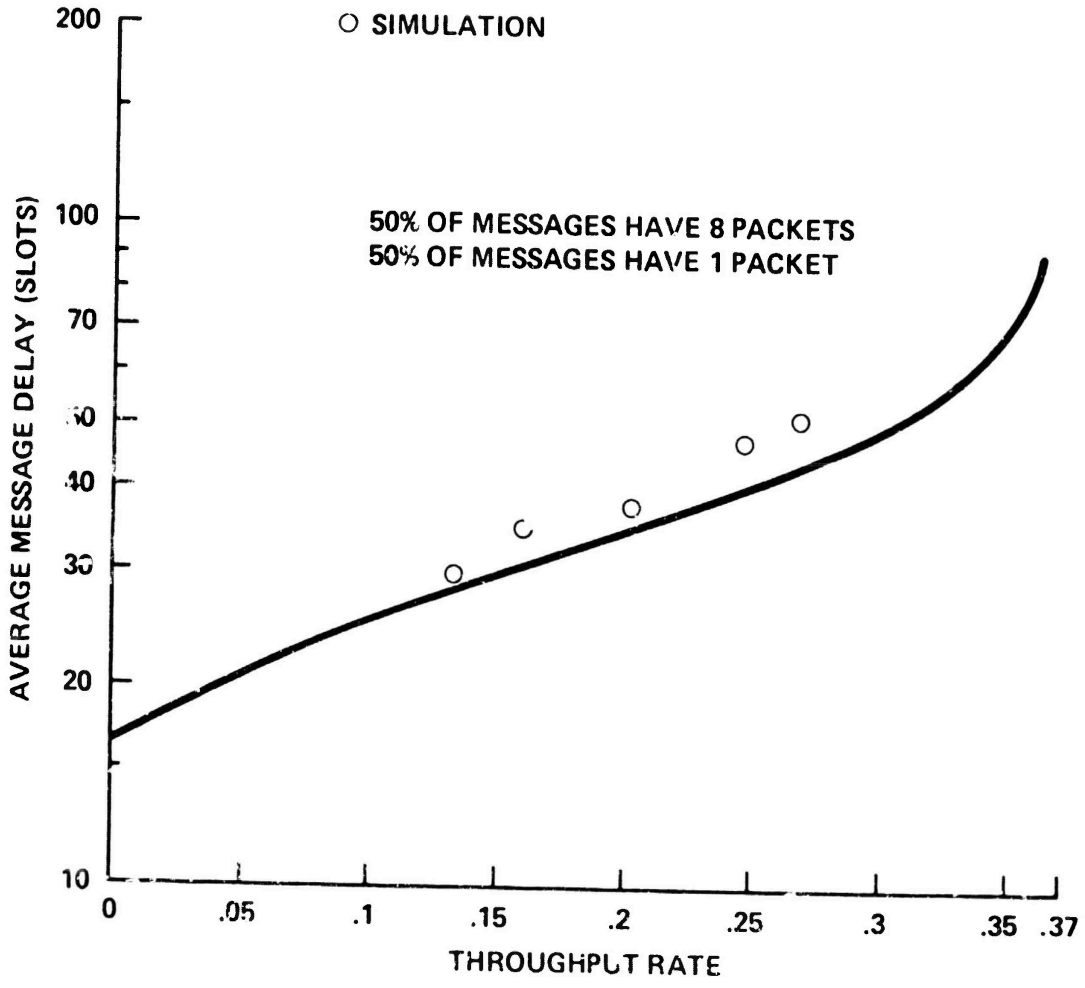


Figure 7-2. Throughput-Delay Tradeoff for Single-Packet and Eight-Packet Messages.

and control analyses in Chapters 5 and 6 to account for multi-packet messages. Such a study, however, is beyond the scope of this dissertation.

7.2 A Reservation System for Multi-Packet Messages

In Roberts' study of a satellite reservation system, the message arrivals to each station are assumed to be Poisson and equally divided between single-packet and eight-packet messages. The channel is dynamically partitioned into a slotted ALOHA reservation channel and a scheduled channel in the following manner [ROBE 73]:

> "....the satellite channel is divided into time slots of 1350 bits each. However, after every M slots one slot is subdivided into V small slots. The small slots are for reservations and acknowledgments, to be used on a contention basis with the ALOHA technique. The remaining M large slots are for RESERVED data packets. When a data packet or multi-packet block arrives at a station it transmits a reservation in a randomly selected one of the V small slots in the next ALOHA group. The reservation is a request for from one to eight RESERVED slots. Upon seeing such a reservation each station adds the number of slots requested to a count, J, the number of slots currently reserved. The originating station has now blocked out a sequence of RESERVED slots to transmit his packets in. Thus, there is one common queue for all stations and by broadcasting reservations they can claim space on the queue. It is not necessary for any station but the originating station to remember which space belongs to whom, since the only requirement is that no one else uses the slots."

Each small ALOHA slot is 224 bits long (V = 6) and can accommodate an acknowledgment packet, a reservation packet or a small data packet. In a reservation packet, the reservation request is triplicated to improve the probability of error-free reception of the request by all stations.

The queueing delay in the above reservation system may be obtained by modelling the common queue as a M/G/1 queueing system

[KLEI 74D]. The delay incurred by a message consists of the slotted ALOHA (reservation) delay and the queueing delay. Thus, the message delay is bounded below by two satellite round-trip propagation times ($\cong$ 0.5 second). Given that the channel input is equally divided between single-packet and eight-packet messages and assuming a 50 KBPS channel and ten stations, Roberts showed that the slotted ALOHA scheme gives a lower average message delay than the reservation system for $S_{out} < 0.15$; however, for $S_{out} > 0.15$, the reservation system gives a lower delay. Furthermore, a channel throughput rate close to one is achievable in the reservation system. For a large population of stations with low data rates, both the slotted ALOHA scheme and the reservation system are far superior to traditional techniques such as Time Division Multiple Access (TDMA) and Frequency Division Multiplexing (FDM). The latter techniques are competitive only when each individual station has a data rate of 50 KBPS. On the other hand, the packet switching techniques depend upon the total multi-station traffic rather than the individual station traffic for their efficiency.

Simulation results for the reservation system indicate that analytic results given by the M/G/1 queueing model are very accurate. However, the slotted ALOHA reservation channel exhibits unstable behavior [LAM 73]. Since the overall performance of the reservation system depends upon the slotted ALOHA reservation channel performance, some dynamic channel control scheme (such as those in Chapter 6) may be necessary.

When a significant fraction of the channel input consists of multi-packet messages, the reservation system has the following advantages compared to the slotted ALOHA scheme:

(1) The average message delay is smaller (except at a low channel throughput rate).

(2) The channel capacity is larger.

(3) The slotted ALOHA scheme tends to be "more unstable" with multi-packet messages. On the other hand, in the reservation system, the slotted ALOHA reservation channel input consists of only single packets (reservation requests, acknowledgments). For a relatively large V (number of small slots in a large slot), a low reservation channel input rate can be maintained for good channel stability using just a small fraction of the total channel bandwidth.

7.3 Reservation-ALOHA Schemes

We describe in this section two satellite "reservation" schemes based upon the slotted ALOHA scheme. By providing some degree of synchronization among the channel users, they are capable of achieving a channel throughput rate well in excess of the slotted ALOHA channel capacity. However, the channel performance of both reservation schemes depends upon (1) a small number of stations (in the order of R, the number of slots in a round-trip satellite propagation time), and (2) each station's input source (of packets) consists of both constant and random components.

The first reservation scheme is known as reservation-ALOHA in which the notion of a "time frame" is introduced [CROW 73]. The channel time is divided into consecutive time frames. Each time frame contains at least R slots. Channel slots in which a station had successful packet transmissions in the previous time frame are reserved for it to use again in the current time frame; no other stations are permitted to use these slots. Channel slots which were either empty or contained collisions in the previous time frame are available for random access by all stations in the current time frame. Thus, once a station has acquired a channel slot, it can keep the same slot in every time frame as long as it has something to transmit. Consequently, the channel performance is very good if the stations have deterministic uniform arrivals; the channel performance suffers when packet arrivals to stations are infrequent and random. Simulation results indicate that a channel throughput rate close to one is achievable at the expense of long packet delays. Also, for a given channel throughput, packet delays tend to increase as the number of stations increases [RETT 73B].

The second reservation scheme to be referred to as priority reservation-ALOHA adds a priority mechanism to the frame principle of reservation-ALOHA [BIND 72]. In this scheme, each station owns at least one slot per frame. The owner of a slot has the highest priority in the event of a collision in the slot. Thus, a station that has been idle is guaranteed channel usage within a maximum of two time frames. Beyond ownership, slots are also assigned. When

two assignees of a slot are involved in a collision, the conflict is resolved by some globally known priority assignment mechanism. Thus, in all cases, each packet requires at most one retransmission. This scheme is more complex to implement than reservation-ALOHA, but promises to give smaller delays to stations with infrequent packet arrivals. The throughput-delay performance of this scheme has not been demonstrated.

CHAPTER 8

CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH

Trends in the growth of computer-communication networks seem to indicate that the next generation of networks will be at least an order of magnitude larger than existing designs. Present implementations, however, are not directly applicable to very large networks. New techniques are needed which can provide cost-effective, high-speed communications for large populations of (potentially mobile) users scattered over wide geographical areas. Under these circumstances, we feel that packet switched satellite and ground radio systems provide attractive solutions to the design of communication subnets and terminal access networks respectively. A packet switching technique which has attracted considerable attention is the slotted ALOHA random access scheme.

The objective of this research was to develop analytic models with which we can evaluate and optimize the performance of a slotted ALOHA channel; our emphasis is on a large population of small users.* Results obtained in this dissertation are summarized in Section 1.4. Major contributions of this research may be classified into three categories:

- a coherent theory of channel behavior in which the key result is the characterization of stable and unstable channels
- evaluation of channel performance such as equilibrium throughput-delay tradeoffs for stable channels and stability-throughput-delay tradeoffs for unstable channels

*Recall that our abstract model of a small user represents a bursty user with buffering space for only one packet.

- dynamic channel control and estimation procedures for optimal control of unstable channels.

To design a slotted ALOHA random access system for small (bursty) users, the following steps may be followed:

(1) Evaluate the equilibrium throughput-delay tradeoff curves. Then, choose an operating value for K (or p) which gives an equilibrium channel throughput-delay tradeoff close to the optimum performance envelope.

(2) Given the average user think time $\frac{1}{\sigma}$, insure that the channel is not overloaded (as shown in Fig. 5-6(d)) by limiting the number of active users (M) who can "sign-on" and use the channel.

(3) For a small enough M, the channel may already be stable according to our stability definition. In this case, the system design is complete.

(4) For bursty users (i.e., $\frac{1}{\sigma}$ is large), a stable channel is associated with a very low channel throughput rate. Increasing M to increase channel utilization will render the channel unstable. In this case, go to either (5) or (6).

(5) If the unstable channel has an acceptable channel failure rate (i.e., FET) or one can be achieved by increasing the operating value of K without significantly increasing delay, the system design is complete. Otherwise, go to (6).

(6) Incorporate into the system (at each channel user) capability for storing channel information within a history window and implementing channel state estimation and dynamic control algorithms. Results in Chapter 6 indicate that with dynamic channel control, a channel throughput-delay performance close to the optimum performance envelope is achievable over an infinite time horizon for (originally) unstable channels.

(7) In a practical system, the load (M) on the channel will probably vary as a function of time with periods of heavy and light loads. The system should be designed for heavily loaded conditions since the performance of a lightly loaded channel is relatively insensitive to the system design. (See, for example, Figures 3-5, 6-12 and 6-13.)

Equilibrium throughput-delay tradeoffs have also been obtained for the large user model and multi-packet messages. In the former case, substantial improvements in the channel performance are possible if the large user accounts for a significant fraction of the channel input rate. In the latter case, if a large fraction of the channel input consists of multi-packet messages and the average message delay is the relevant performance measure, we concluded that Roberts' reservation system is superior to slotted ALOHA. Note, however, that the reservation system utilizes the slotted ALOHA scheme for broadcasting reservation requests; our slotted ALOHA results apply to the reservation subchannel in this system.

Numerical results in this dissertation were obtained assuming a 50 KBPS satellite channel with 1125 bits/packet and a channel round-trip propagation delay of 0.27 second. The models and methodology

developed, however, are independent of these assumptions and may be applied to satellite channels with different data rates, ground radio systems as well as wire communications such as multi-drop lines and loop systems.

Extensions to this research

Before small satellite earth systems become an economic reality, satellite channel users will tend to be "big" and few in number. For example, the Satellite IMP, being designed for the ARPANET Satellite System, will have buffer space for 32 packets [BUTT 74]. This situation corresponds to the finite population model studied in Chapter 3. Our stability and dynamic channel control results in Chapters 5 and 6 may be extended to this case. However, the state description is now a vector consisting of the queue sizes at all satellite stations instead of a single variable such as i in the linear feedback model.

For dynamic channel control procedures considered in this research, optimal control policies were found to be of the control limit type in all our numerical examples. A rigorous mathematical proof of this result remains an open problem.

The slotted ALOHA channel is characterized by the throughput-load curve depicted in Fig. 8-1, which is typical of "contention" systems [AGNE 73]. Unlike queueing systems in which the throughput increases to one as the system load increases, the throughput of a contention system increases to a maximum value and then decreases.

In this dissertation, we have characterized the unstable behavior and studied dynamic control schemes for a specific contention system,

namely, slotted ALOHA random access. The probabilistic model and techniques employed here can probably be extended to solve stability and dynamic control problems of other contention systems.

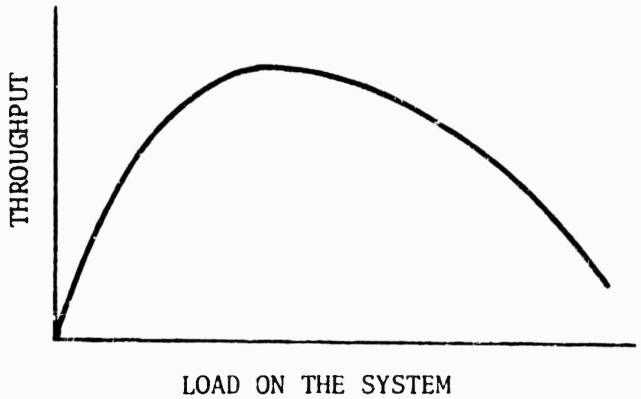


Fig. 8-1 A Typical Throughput-Load Curve for a Contention System

One class of contention systems consists of random access packet switching techniques (relatives of slotted ALOHA!) such as pure ALOHA, FM capture and carrier sense. These systems seem to exhibit unstable behavior similar to that of slotted ALOHA and may be dynamically controlled by similar schemes. As of now, most efforts in the study of these systems have been concentrated on the evaluation of the system capacity and equilibrium throughput-delay tradeoff. Little attention has been given to the problems of stability and control. For example, the ALOHA System at the University of Hawaii has been estimated to be able to support over 300 interactive users (assuming the multi-access channel capacity to be $\frac{1}{2e} \cong 18\%$) [ABRA 70]. We feel that these figures are unrealistic for an uncontrolled system, but may be achieved given appropriate dynamic channel control.

Many other existing systems can also be characterized as contention systems and exhibit unstable behavior similar to that of slotted ALOHA. A highway is a contention system and Fig. 8-1 represents the so-called "fundamental diagram of traffic" [ASHT 66]. Simulation results for store-and-forward packet switching networks show that they have throughput-load curves similar to that depicted in Fig. 8-1 [KAHN 71, DAVI 71]. It is interesting to note that heuristic flow control routing algorithms suggested by Kahn and Crowther [KAHN 71] and the so-called "isarithmic" networks proposed by Davies [DAVI 71] are similar in spirit to our dynamic channel control procedures.

Agnew considered a general deterministic model of a contention system and studied its dynamic control through pricing [AGNE 73]. A topic of future research interest is the formulation of a general probabilistic model of a contention system. It may be possible to extend the stability and dynamic control results in this dissertation to the general model.

BIBLIOGRAPHY

ABRA 70 Abramson, N. "THE ALOHA SYSTEM--Another Alternative for Computer Communications," Fall Joint Computer Conference, AFIPS Conference Proceedings, 1970, Vol. 37, pp. 281-285.

ABRA 72 Abramson, N. "ARPANET Satellite System," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 7 (NIC 11289), May 1972.

ABRA 73 Abramson, N. "Packet Switching with Satellites," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, 1973, Vol. 42, pp. 695-702.

AGNE 73 Agnew, C.E. "The Dynamic Control of Congestion-Prone Systems through Pricing," Center for Interdisciplinary Research, Stanford University, Stanford, California, Report No. 6, November 1973.

ASHT 66 Ashton, W.D. The Theory of Road Traffic Flow, Methuen, London, 1966.

BARA 64 Baran, P. "On Distributed Communications: XI. Summary Overview," Rand Corporation, Santa Monica, California, Memorandum RM-3767-PR, August 1964.

BAYL 73 Bayless, J.W., S.J. Campanella and A. J. Goldberg. "Voice Signals: Bit-by-Bit," IEEE Spectrum, Vol. 10, No. 10, pp. 28-36, October 1973.

BEVE 70 Beveridge, G.S. and R.S. Schechter. Optimization: Theory and Practice, McGraw-Hill, New York, 1970.

BIND 72 Binder, R. "Another ALOHA Satellite Protocol," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 32 (NIC 13147), December 1972.

BUTT 74 Butterfield, S., R. Rettberg and D. Walden. "The Satellite IMP for the ARPA Network," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

CACC 71 Cacciamani, E. "The SPADE System as Applied to Data Communications and Small Earth Station Operation," COMSAT Technical Review, Vol. 1, No. 1, pp. 171-182, 1971.

CACC 74 Cacciamani, E. "Data Services--American Satellite Corporation," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

CARR 70 Carr, S., S. Crocker and V. Cerf. "HOST-HOST Communication Protocol in the ARPA Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1970, Vol. 36, pp. 589-597.

CHU 69 Chu, W.W. "A Study of Asynchronous Time Division Multiplexing for Time-Sharing Computer Systems," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1969, Vol 35, pp. 669-678.

CHUN 67 Chung, K.L. Markov Chains with Stationary Transition Probabilities, Springer-Verlag, New York, 1967.

COHE 69 Cohen, J.W. The Single Server Queue, Wiley. New York, 1969.

CRAI 64 Craig, E.J. Laplace and Fourier Transforms for Electrical Engineers, Holt, Rinehart and Winston, New York, 1964.

CROC 72 Crocker, S., J. Heafner, R. Metcalfe and J. Postel. "Function-Oriented Protocols for the ARPA Computer Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1972. Vol. 40, pp. 271-279.

CROW 73 Crowther, W., R. Rettberg, D. Walden, S. Ornstein, and F. Heart. "A System for Broadcast Communication: Reservation-ALOHA," Proceedings of the Sixth Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 1973.

DAVI 68 Davies, D.W. "The Principles of a Data Communication Network for Computers and Remote Peripherals," Proceedings IFIP Congress, Edinburgh, Scotland, 1968, pp. D11-D15.

DAVI 71 Davies, D.W. "The Control of Congestion in Packet Switching Networks," Proceedings of the Second ACM/IEEE Symposium on Problems in the Optimization of Data Communications Systems, 1971, pp. 46-49.

DUNN 74 Dunn, D. and M. Eric. "The Economics of Packet Switching," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

FELL 68 W. An Introduction to Probability Theory and Its Applications, 3rd ed., Vol. I, Wiley, New York, 1968.

FRAN 70 Frank, H., I. Frisch and W. Chou. "Topological Considerations in the Design of the ARPA Computer Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1970, Vol. 36, pp. 581-587.

FRAN 72A Frank, H., R. Kahn and L. Kleinrock. "Computer Communication Network Design--Experience with Theory and Practice," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1972, Vol. 40, pp. 255-270.

FRAN 72B Frank, H. and W. Chou. "Topological Optimization of Computer Networks," Proceedings of the IEEE, Vol. 60, No. 11, pp. 1385-1396, November 1972.

FRAN 73 Frank, H., M. Gerla and W. Chou. "Issues in the Design of Large Distributed Computer Communication Networks," Proceedings of the National Telecommunication Conference, Atlanta, November 26-28, 1973.

FUCH 70 Fuchs, E. and P.E. Jackson. "Estimates of Distributions of Random Variables for Certain Computer Communications Traffic Models," Communications of the ACM, Vol. 13, No. 12, pp. 752-757, December 1970.

FULT 72 Fultz, G.L. "Adaptive Routing Techniques for Message Switching Computer-Communication Networks," School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7252, July 1972.

GAAR 72 Gaarder, N.T. "ARPANET Satellite System," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 3 (NIC 11285), April 1972.

GABB 68 Gabbard, O. "Design of a Satellite Time-Division Multiple-Access Burst Synchronizer," IEEE Transactions on Communications Technology, Vol. COM-16, No. 4, pp. 589-596, August 1968.

GERL 73 Gerla, M., W. Chou and H. Frank. "Computational Considerations and Routing Problems for Large Computer Communication Networks," Proceedings of the National Telecommunication Conference, Atlanta, November 26-28, 1973.

GITM 74 Gitman, I., R. Van Slyke and H. Frank. "On Splitting Random Accessed Broadcast Communication Channels," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

GRAY 74 Gray, E.M. "ANIK," Communications Society, Vol. 12, No. 1, pp. 3-5, January 1974.

HAYE 71 Hayes, J. and D. Sherman. "Traffic Analysis of a Ring Switched Data Transmission System," Bell System Technical Journal, Vol. 50, No. 9, pp. 2947-2978, November 1971.

HAYE 72 Hayes, J. and D. Sherman. "A Study of Data Multiplexing Techniques and Delay Performance," The Bell System Technica Journal, Vol. 51, No. 9, pp. 1983-2011, November 1972.

HEAR 70 Hear[illegible], F., R. Kahn, S. Ornstein, W. Crowther and D. Walden. "The Interface Message Processor for the ARPA Computer Network," Spring Joint Compu[illegible]er Conference, AFIPS Conference Proceedings, 1970, Vol. 36, pp. 551-567.

HOWA 60 Howard, R. Dynamic Programming and Markov Processes, M.I.T. Press, Cambridge, Mass., 1960.

HOWA 71 Howard, R. Dynamic Probabilistic Systems, Vol. 1: Markov Models and Vol. 2: Semi-Markov and Decision Processes, Wiley, New York, 1971.

JACK 69 Jackson, P.E. and C.D. Stub[illegible]s. "A Study of Multiaccess Computer Communications," Spr[illegible]g Joint Computer Conference, AFIPS Conference Proceedings, 1969, Vol. 34, pp. 491-504.

KAHN 71 Kahn, R.E. and W.R. Crowther. "Flow Control in a Resource-Sharing Computer Network," Proceedings of the Second ACM/IEEE Symposium on Problems in the Optimization of Data Communications Systems, 1971, pp. 108-116.

KLEI 64 Kleinrock, L. Communication Nets: Stochastic Message Flow and Delay, McGraw Hill, New York, 1964, out of print (Reprinted by Dover Publications, New York, 1972).

KLEI 70 Kleinrock, L. "Analytic and Simulation Methods in Computer Network Design," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1970, Vol. 36, pp. 569-579.

KLEI 72A Kleinrock, L. and S.S. Lam. "Analytic Results for the ARPANET Satellite System Model Including the Effects of the Retransmission Delay Distribution," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 12 (NIC 11294), August 1972.

KLEI 72B Kleinrock, L. and S.S. Lam. "Approximations in the Infinite Population Model of the ARPANET Satellite System," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 17 (NIC 11862), October 1972.

KLEI 72C Kleinrock, L. and S.S. Lam. "Correction for ASS Note 12," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 25 (NIC 12734), November 1972.

KLEI 72D Kleinrock, L. and S.S. Lam. "Analytic Results with the Addition of One Large User," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 27 (NIC 12736), October 1972.

KLEI 73A Kleinrock, L. and S.S. Lam. "Packet-Switching in a Slotted Satellite Channel," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, 1973, Vol. 42, pp. 703-710.

KLEI 73B Kleinrock, L. and S.S. Lam. "Dynamics of the ALOHA Channel," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 50 (NIC 18455), August 1973.

KLEI 74A Kleinrock, L. and S.S. Lam. "On Stability of Packet Switching in a Random Multi-Access Broadcast Channel," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

KLEI 74B Kleinrock, L., and W.E. Naylor. "On Measured Behavior of the ARPA Network," to be presented at the National Computer Conference, Chicago, May 1974.

KLEI 74C Kleinrock, L. and F.A. Tobagi. "Carrier-Sense Multiple Access for Packet Switched Radio Channels," to be presented at the International Conference on Communications, Minneapolis, Minn., June 1974.

KLEI 74D Kleinrock, L., Queueing Systems, Vol. I, Theory, Vol. II, Computer Applications, to be published by Wiley Interscience, New York, 1974.

KUO 73 Kuo, F.F., and N. Abramson. "Some Advances in Radio Communications for Computers," Digest of papers, Seventh Annual IEEE Computer Society International Conference, San Francisco, California, February 1973, pp. 57-60.

LAM 73 Lam, S.S. "Some Satellite Simulation Results," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 48 (NIC 17655), July 1973.

LITT 61 Little, J. "A Proof of the Queuing Formula $L = \lambda W$," Operations Research, Vol. 9, No. 2, pp. 383-387, March-April 1961.

LU 73 Lu, S. "Dynamic Analysis of Slotted ALOHA with Blocking," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 36 (NIC 14790), March 1973.

MART 71 Martin, J. Future Developments in Telecommunications, Prentice-Hall, Englewood Cliffs, N.J., 1971.

MART 72 Martin, J. Systems Analysis for Data Transmission, Prentice-Hall, Englewood Cliffs, N.J., 1972.

METC 73A Metcalfe, R.M. "Steady-State Analysis of a Slotted and Controlled ALOHA System with Blocking," Proceedings of the Sixth Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 1973.

METC 73B Metcalfe, R.M. "Packet Communication," Project MAC, Massachusetts Institute of Technology, Cambridge, Mass., Report MAC TR-114, July 1973.

NAC 73 Network Analysis Corporation. "The Practical Impact of Recent Computer Advances on the Analysis and Design of Large Scale Networks," First Semiannual Technical Report, Glen Cove, New York, May 1973.

NEWE 68 Newell, G.F. "Queues with Time-Dependent Arrival Rates I--the Transition through Saturation," Journal of Applied Probability, Vol. 5, No. 2, pp. 436-451, August 1968.

ORNS 72 Ornstein, S., F. Heart, W. Crowther, H. Rising, S. Russell and A. Michel. "The Terminal IMP for the ARPA Computer Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1972, Vol. 40, pp. 243-254.

PARZ 62 Parzen, E. Stochastic Processes, Holden-Day, San Francisco, 1962.

PIER 71 Pierce, J.R. "Network for Block Switching of Data," IEEE Convention Record, New York, March 1971.

PUEN 71 Puente, J. and A. Werth. "Demand Assignment Service for the INTELSAT Global Network," IEEE Spectrum, Vol. 8, No. 1, pp. 56-69, January 1971.

RETT 72 Rettberg, R. "A Brief Simulation of the Dynamics of an ALOHA System with Slots," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 11 (NIC 11293), July 1972.

RETT 73A Rettberg, R. "Slotting," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 42 (NIC 16085), May 1973.

RETT 73B Rettberg, R. "Preliminary Simulation Results for Reservation ALOHA," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 43 (NIC 16086), May 1973.

RETT 73C Rettberg, R. "Geometric Retransmission Randomization," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 51 (NIC 18744), September 1973.

ROBE 70 Roberts, L.G. and B.D. Wessler. "Computer Network Development to Achieve Resource Sharing," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1970, Vol. 36, pp. 543-549.

ROBE 72A Roberts, L.G. "Extensions of Packet Communication Technology to a Hand Held Personal Terminal," Spring Joint Computer Conference, AFIPS Conference Proceedings, 1972, Vol. 40, pp. 295-298.

ROBE 72B Roberts, L.G. "ALOHA Packet System With and Without Slots and Capture," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 8 (NIC 11290), June 1972.

ROBE 73 Roberts, L.G. "Dynamic Allocation of Satellite Capacity through Packet Reservation," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, 1973, Vol. 42, pp. 711-716.

ROBE 74 Roberts, L.G. "Data by the Packet," IEEE Spectrum, Vol. 11, No. 2, pp. 46-51, February 1974.

ROSS 70 Ross, S.M. Applied Probability Models with Optimization Applications, Holden-Day, San Francisco, 1970.

RUDI 66 Rudin, W. Real and Complex Analysis, McGraw-Hill, New York, 1966.

SCHW 65 Schwarz, R.J. and B. Friedland. Linear Systems, McGraw-Hill, New York, 1965.

TELE 73 Telenet Communications Corporation. "Before the Federal Communications Commission, Application for Authority to Establish and Operate a Packet-Switched Data Communications Network in the Continental United States," Washington, D.C., October 1973.

WRIG 73 Wright, P. "Facing a Booming Demand for Networks," Datamation, Vol. 19, No. 11, pp. 138-139, November 1973.

APPENDIX A

SIMULATION RESULTS FOR THE POISSON ASSUMPTION

Channel traffic is a random variable representing the total number of packets transmitted by all users into a channel time slot. Both zeroth order and first order approximations in Chapter 3 assume that channel traffic is Poisson distributed (the Poisson assumption). In this appendix, we examine further the accuracy of the Poisson assumption through simulations.

Let P_i be the fraction of time slots, each of which has exactly i packet transmissions, over the duration of a simulation run. $\left| P_i \right|_{i=0}^{M}$ represents the measured probability distribution for channel traffic. (M is the number of channel users.) The channel throughput rate S_{out} is given by P_1. The channel traffic rate G is given by $\sum_{i=1}^{M} i P_i$. We give below comparisons between P_i and the Poisson probabilities $\frac{G^i}{i!} e^{-G}$ for the infinite population model, the linear feedback model and controlled channels.

In Table A.1, P_i and the corresponding Poisson probabilities are shown for various cases of the infinite population model. In all cases, $R = 12$ and the simulation duration is 8000 time slots. Each simulation run satisfies the channel equilibrium criterion in Section 3.2.3. Cases (a), (b) and (c) correspond to $K = 5$, 15 and 40 respectively with $S_{out} \cong 0.25$. Note that the Poisson approximation is better for $K = 15, 40$ than $K = 5$. (This observation is consistent with the conclusion of Theorem 4.1.) Cases (b), (d) and (e) correspond to $S_{out} = 0.245$, 0.150 and 0.304 respectively with $K = 15$. Note that the Poisson approximation is better for a smaller S_{out}.

In Table A.2, comparisons are shown for the linear feedback model with $M = 200$ and four different retransmission delay probability distributions (corresponding to those in Fig. 5-1). Each simulation run has a duration of 8000 time slots and satisfies the channel equilibrium criterion. In all four cases, the Poisson approximation is excellent.

In Table A.3, comparisons are shown for three controlled channels with $M = 200$: (a) ICP-CONTEST with $W = 40$ and $\hat{n} = 22$, (b) RCP-CONTEST with $W = 40$ and $\hat{n} = 18$, and (c) Heuristic RCP with $K_1 = 10$ and $K_m = 60$ for $m \geq 2$. R is assumed to be 12 and each simulation run has a duration of 30,000 time slots. In all cases the Poisson approximation is quite good. (Note that performance of the CONTEST algorithms depends upon the accuracy of $P_0 \cong e^{-G}$ within a time history window.)

From comparisons in Tables A.1 - A.3, we also observe the following:

(1) In all cases, $P_0 \geq e^{-G}$

(2) In all cases, $P_1 \leq Ge^{-G}$; this is expected since finite retransmission delays are used.

(3) In most cases, P_i $(2 \leq i \leq 6)$ are larger than the corresponding Poisson probabilities. On the other hand, the Poisson distribution has a much longer tail than the measured channel traffic probability distribution.

(a) K = 5

S_{out} = 0.253

G = 0.432

i	P_i	Poisson
0	0.6671	0.6490
1	0.2530	0.2806
2	0.0649	0.0607
3	0.0116	0.0087
4	0.0025	0.0009
5	0.0005	0.0001
6	0.0004	0.0000

(b) K = 15

S_{out} = 0.245

G = 0.361

i	P_i	Poisson
0	0.7011	0.6973
1	0.2451	0.2514
2	0.0466	0.0453
3	0.0064	0.0054
4	0.0008	0.0005
5	0.0000	0.0000
6	0.0000	0.0000

(c) K = 40

S_{out} = 0.252

G = 0.784

i	P_i	Poisson
0	0.6872	0.6814
1	0.2516	0.2614
2	0.0524	0.0501
3	0.0080	0.0064
4	0.0008	0.0006
5	0.0000	0.0000
6	0.0000	0.0000

Table A.1 Channel traffic probability distribution (infinite population model).

(d) K = 15

S_{out} = 0.150

G = 0.184

i	P_i	Poisson
0	0.8335	0.8315
1	0.1500	0.1534
2	0.0153	0.0142
3	0.0011	0.0009
4	0.0000	0.0000
5	0.0001	0.0000
6	0.0000	0.0000

(e) K = 15

S_{out} = 0.304

G = 0.586

i	P_i	Poisson
0	0.5722	0.5563
1	0.3045	0.3263
2	0.0946	0.0957
3	0.0229	0.0187
4	0.0048	0.0027
5	0.0009	0.0003
6	0.0001	0.0000

Table A.1 (Continued)

i	P_i	Poisson
0	0.6086	0.6021
1	0.2944	0.3055
2	0.0805	0.0775
3	0.0141	0.0131
4	0.0023	0.0016
5	0.0001	0.0002
6	0.0000	0.0000

(a) R = 12 K = 10

S_{out} = 0.294

G = 0.507

i	P_i	Poisson
0	0.6264	0.6256
1	0.2934	0.2934
2	0.0670	0.0688
3	0.0116	0.0108
4	0.0014	0.0013
5	0.0002	0.0001
6	0.0000	0.0000

(b) R = 0 K = 34

S_{out} = 0.293

G = 0.469

i	P_i	Poisson
0	0.6308	0.6279
1	0.2894	0.2922
2	0.0661	0.0680
3	0.0113	0.0105
4	0.0023	0.0012
5	0.0001	0.0001
6	0.0000	0.0000

(c) R = 12 $p = \frac{2}{11}$

S_{out} = 0.289

G = 0.465

i	P_i	Poisson
0	0.6373	0.6351
1	0.2831	0.2883
2	0.0691	0.0655
3	0.0095	0.0099
4	0.0009	0.0011
5	0.0001	0.0001
6	0.0000	0.0000

(d) R = 0 $p = \frac{2}{35}$

S_{out} = 0.283

G = 0.454

Table A.2 Channel traffic probability distribution (M = 200).

(a) ICP-CONTEST

$K = 10$

$S_{out} = 0.315$

$G = 0.597$

i	P_i	Poisson
0	0.5612	0.5505
1	0.3148	0.3286
2	0.0963	0.0981
3	0.0223	0.0195
4	0.0044	0.0029
5	0.0009	0.0004
6	0.0001	0.0000

(b) RCP-CONTEST

$K_o = 10$ $K_c = 60$

$S_{out} = 0.322$

$G = 0.655$

i	P_i	Poisson
0	0.5340	0.5193
1	0.3218	0.3403
2	0.1084	0.1115
3	0.0282	0.0243
4	0.0061	0.0040
5	0.0014	0.0005
6	0.0000	0.0001

(c) Heuristic RCP

$K_1 = 10$ $K_2 = 60$

$S_{out} = 0.316$

$G = 0.579$

i	P_i	Poisson
0	0.5670	0.5605
1	0.3163	0.3245
2	0.0922	0.0939
3	0.0205	0.0181
4	0.0034	0.0026
5	0.0005	0.0003
6	0.0001	0.0000

Table A.3 Channel traffic probability distribution (Controlled Channels).

APPENDIX B

ANALYSIS FOR THE LARGE USER MODEL

The set of nonlinear implicit equations in solving equilibrium values of S_i, G_i, q_{in} and q_{it} (i = 1, 2) in the large user model will be derived. Recall that variables indexed by 1 refer to the small users and variables indexed by 2 refer to the large user.

Define E_1 and E_2 to be the average number of channel collisions for a small user and a large user packet respectively. Similar to the derivation of Eq. (3.5), we have

$$E_i = (1 - q_{in})/q_{it} \qquad i = 1, 2 \qquad \text{(B.1)}$$

$$G_i = S_i(1 + E_i) \qquad i = 1, 2 \qquad \text{(B.2)}$$

Thus

$$S_i = G_i \frac{q_{it}}{q_{it} + 1 - q_{in}} \qquad i = 1, 2 \qquad \text{(B.3)}$$

which are Eqs. (3.16) and (3.17).

Referring to the model description of a large user in Section 2.3.2, we introduce the following notation for events at the large user:

TS = transmission success in a channel slot

SS = scheduling success (i.e., capture of the transmitter as a result of having the highest priority among all packets scheduled for the current time slot)

Each large user packet may be in one of the following three states depending on their most recent history:

NP = newly generated packet

SC = scheduling conflict (i.e., failure to capture transmitter)

TC = transmission conflict in a channel slot

Now define the variables,

a_n = Prob [SS/NP]

a_t = Prob [SS/TC]

a_s = Prob [SS/SC]

r_n = Prob [TS/SS, NP]

r_t = Prob [TS/SS, TC]

r_s = Prob [TS/SS, SC]

Given a large user packet, let E_n and E_t be the average number of SC events before SS, conditioning on NP and TC respectively. Similar to the derivation of Eq. (3.5), we have

$$E_n = (1 - a_n)/a_s$$
$$E_t = (1 - a_t)/a_s \qquad \text{(B.4)}$$

Recalling the definitions of q_{2n} and q_{2t}, we have

$$q_{2n} = (r_n + r_s E_n)/(1 + E_n)$$
$$q_{2t} = (r_t + r_s E_t)/(1 + E_t) \qquad \text{(B.5)}$$

The average station traffic (defined in Section 3.3.2) is

$$G_s = S_2 [1 + E_n + E_2(1 + E_t)] \qquad \text{(B.6)}$$

and the average packet delays are

$$D_1 = R + 1 + E_1\left[R + \frac{K+1}{2}\right] \tag{3.18}$$

$$D_2 = R + 1 + E_2\left[R + \frac{K+1}{2}\right] + (E_n + E_2E_t)\frac{L+1}{2} \tag{3.19}$$

where $R + \frac{K+1}{2}$ is the average retransmission delay and $\frac{L+1}{2}$ is the average reschedule delay (see Section 2.3.2).

With the Poisson and independence assumptions in Section 3.3.2 for channel traffic and station traffic, we proceed to solve for the success probabilities q_1, q_{1t}, r_n, r_t, r_s, a_n, a_t and a_s. (The approach is similar to the derivation of q_n and q_t in the infinite population model.) Consider the transmission of a test packet in the current time slot; a conflict may occur as a result of new arrivals, packets retransmitting from a window of K slots or packets rescheduling from a window of L slots in the past.

Define

q_o = Prob [no packet retransmitting from one of the K slots to the current slot]

and

q_h = Prob [no packet rescheduling from one of the L slots to the current slot]

We then have,

$$q_o = \sum_{n,m\geq 1} \frac{G_1^n}{n!} e^{-G_1} \frac{G_s^m}{m!} e^{-G_s\left(\frac{K-1}{K}\right)^{n+1}} + e^{-G_1} \sum_{m\geq 1} \frac{G_s^m}{m!} e^{-G_s} +$$

$$e^{-G_s}\sum_{n\geq 2}\frac{G_1^n}{n!}e^{-G_1}\left(\frac{K-1}{K}\right)^n + G_1 e^{-(G_1+G_s)} + e^{-(G_1+G_s)}$$

$$= e^{-G_1/K} + \frac{1}{K}\left[\left(1-e^{-G_s}\right)\left(e^{-G_1}-e^{-G_1/K}\right) + G_1 e^{-(G_1+G_s)}\right]$$

$$q_h = \sum_{m\geq 2}\frac{G_s^m}{m!}e^{-\left(\frac{L-1}{L}\right)^{m-1}} + G_s e^{-G_s} + e^{-G_s}$$

$$= \left[L\, e^{-G_s/L} - e^{-G_s}\right]\Big/(L-1) \qquad L\geq 2$$

$$q_h = (G_s+1)\, e^{-G_s} \qquad L = 1$$

Suppose the test packet is a small user packet. Conditioning on a new packet, we have

$$q_{1n} = q_o^K q_h^L e^{-S} \tag{B.7}$$

Conditioning on a packet which had a channel collision in the j^{th} slot, define

q_{1c} = Prob [no other packet retransmitting from the j^{th} slot to the current slot]

$$q_{1c} = \frac{1}{1-e^{-(G_1+G_s)}}\left[\left(1-e^{-G_s}\right)\sum_{n\geq 0}\frac{G_1^n}{n!}e^{-G_1}\left(\frac{K-1}{K}\right)^{n+1} + e^{-G_s}\sum_{n\geq 1}\frac{G_1^n}{n!}e^{-G_1}\left(\frac{K-1}{K}\right)^n\right]$$

$$= \left[e^{-G_1/K}\left(1-\frac{1-e^{-G_s}}{K}\right) - e^{-(G_1+G_s)}\right]\Big/\left[1-e^{-(G_1+G_s)}\right]$$

We have,

$$q_{1t} = q_o^{K-1} q_{1c} q_h^{L} e^{-S} \tag{B.8}$$

Suppose the test packet is a large user packet and condition on the event SS. Define

$\bar{q}$ = Prob [no small user packet retransmitting from one of the K slots to the current slot]

$$\bar{q} = \sum_{m+n\geq 2} \frac{G_1^n}{n!} e^{-G_1} \frac{G_s^m}{m!} e^{-G_s} \left(\frac{K-1}{K}\right)^n + (G_1+G_s) e^{-(G_1+G_s)} + e^{-(G_1+G_s)}$$

$$= e^{-G_1/K} + \frac{G_1}{K} e^{-(G_1+G_s)}$$

Conditioning on the event NP, we have

$$r_n = \text{Prob [TS/SS, NP]} = \bar{q}^K e^{-S_1} \tag{B.9}$$

and

$$\text{Prob [SS, TS/NP]} = q_o^K q_h^L e^{-S_1} (1-e^{-S_2})/S_2$$

where we have made use of the scheduling priority rule in Section 2.3.2 in which new packets have the lowest priority; ties among new packets are broken by random selection such that

$$(1-e^{-S_2})/S_2 = \sum_{i=0}^{\infty} \frac{S_2^i}{i!} e^{-S_2} \left(\frac{1}{i+1}\right)$$

Finally,

$$a_n = \text{Prob}\ [SS/NP]$$

$$= \text{Prob}\,[SS,\ TS/NP]/\text{Prob}\,[TS/SS,\ NP]$$

$$= \left(q_o/\bar{q}\right)^K q_h^{\,L} \left(1-e^{-S_2}\right)/S_2 \qquad \text{(B.10)}$$

Given that the large user test packet had a channel collision (TC) in the j^{th} slot, define

q_{2c} = Prob [no small user packet retransmitting from the j^{th} slot to the current slot]

$$q_{2c} = \left[\sum_{n\geq 1} \frac{G_1^{\,n}}{n!}\, e^{-G_1} \left(\frac{K-1}{K}\right)^n\right] \Bigg/ \left[1-e^{-G_1}\right]$$

$$= \left[e^{-G_1/K} - e^{-G_1}\right] \Big/ \left[1-e^{-G_1}\right]$$

We then have

$$r_t = \text{Prob}\ [TS/SS,\ TC]$$

$$= \bar{q}^{\,K-1}\, q_{2c}\, e^{-S_1} \qquad \text{(B.11)}$$

and

$$\text{Prob}\ [SS,\ TS/TC] = \sum_{i=1}^{K} \frac{1}{K}\, q_o^{\,i-1} q_{2c}\, \bar{q}^{K-i} e^{-S_1}$$

where the scheduling priority rule has been used.

Finally

$$a_t = \text{Prob [SS/TC]}$$

$$= \text{Prob [SS, TS/TC]} / \text{Prob [TS/SS, TC]}$$

$$= \frac{1}{K} \left[1 - \left(q_o / \bar{q} \right)^K \right] \Big/ \left[1 - \left(q_o / \bar{q} \right) \right] \qquad \text{(B.12)}$$

Given that the large user packet had a scheduling conflict (SC) in the j^{th} slot, define

q_{sc} = Prob [no other packet rescheduling from the j^{th} slot to the current slot]

$$q_{sc} = \left[\sum_{m=1}^{\infty} \frac{G_s^m}{m!} e^{-G_s} \frac{m}{m+1} \left(\frac{L-1}{L} \right)^{m-1} \right] \Big/ \left[\sum_{m=1}^{\infty} \frac{G_s^m}{m!} e^{-G_s} \frac{m}{m+1} \right]$$

$$= \left(\frac{L}{L-1} \right)^2 \left[\frac{G_s \left(1 - \frac{1}{L} \right) e^{-G_s/L} - e^{-G_s/L} + e^{-G_s}}{G_s - 1 + e^{-G_s}} \right]$$

We have

$$r_s = \text{Prob [TS/SS, SC]} \qquad \text{(B.13)}$$

$$= \bar{q}^K e^{-S_1}$$

Prob [SS, TS/SC]

$$= q_o^K q_{sc} \sum_{i=1}^{L} \frac{1}{L} q_n^{i-1} e^{-S_1}$$

where the scheduling priority rule has been used. Finally

$$a_s = \text{Prob}\,[SS/SC]$$

$$= \text{Prob}[SS,\ TS/SC]/\text{Prob}[TS/SS,\ SC]$$

$$= \left(q_o/\bar{q}\right)^K \frac{q_{sc}}{L}\ \frac{1-q_h^L}{1-q_h} \qquad \text{(B.14)}$$

Eqs. (B.3) - (B.14) constitute a set of nonlinear implicit equations which may be solved numerically with specified values of K, L, G_1, and G_2 (or S_1 and S_2).

Limiting results

In the limit as K, $L \to \infty$, the following limiting values may be obtained from the definitions of q_o, q_h, $\bar{q}$, q_{1c}, q_{2c} and q_{sc}:

$$q_o^K = e^{-G_1\left(1-e^{-(G_1+G_s)}\right) - \left(1-e^{-G_1}\right)\left(1-e^{-G_s}\right)}$$

$$q_h^L = e^{-G_s+1-e^{-G_s}}$$

$$\bar{q}^K = e^{-G_1\left(1-e^{-(G_1+G_s)}\right)}$$

$$q_{1c} = q_{2c} = q_{sc} = 1$$

With the above limiting results, the following proposition may be shown.

Proposition B.1 In the limit as K, L $\to \infty$,

$$q_{1n} = q_{1t} = e^{-G_1(1-G_2)} \tag{3.20}$$

$$S_1 = G_1 e^{-G_1(1-G_2)} \tag{3.21}$$

$$q_{2n} = q_{2t} = e^{-G_1} \tag{3.22}$$

$$S_2 = G_2 e^{-G_1} \tag{3.23}$$

$$G_2 = 1 - e^{-G_s} \tag{3.24}$$

$$r_n = r_t = r_s = e^{-G_1}$$

$$a_n = e^{-(G_s-S_2)} \left| 1 - e^{-S_2} \right| \Big/ S_2$$

$$a_t = \left| 1 - e^{-(G_2-S_2)} \right| \Big/ (G_2-S_2)$$

$$a_s = e^{-(G_2-S_2)} \left| 1 - e^{-(G_s-G_2)} \right| \Big/ (G_s-G_2)$$

Proof The variables in the above equations are defined by Eqs. (B.3) - (B.14). It suffices to show that limiting values of these variables given by the proposition satisfy Eqs. (B.3) - (B.14) in the K, L $\to \infty$ limit. This may be accomplished by assuming the proposition to be true, evaluating the RHSs of Eqs. (B.3) - (B.14) and showing that they are equal to the corresponding LHSs in the K, L $\to \infty$ limit.

APPENDIX C

DERIVATION OF EQS. (4.3) AND (4.4), THEOREM 4.1

AND ITS PROOF

Derivation of Eq. (4.3)

By definition,

$$Q^{t+1}(\underline{z}) = \sum_{y_1=0}^{\infty} \cdots \sum_{y_{R+K}=0}^{\infty} \left(\prod_{j=1}^{R+K} z_j^{y_j} \right) p^{t+1}(\underline{y})$$

Substituting Eqs. (4.1) and (4.2) for $p^{t+1}(\underline{y})$, we have

$$Q^{t+1}(\underline{z}) = \sum_{y_1=0}^{\infty} \cdots \sum_{y_{R+K}=0}^{\infty} \left(\prod_{j=1}^{R+K} z_j^{y_j} \right) \sum_{x_{R+K}=0}^{\infty} \sum_{\substack{i=0 \\ i \le \ell}}^{y_1} v_{y_1-i}^{t+1} \binom{\ell}{i} \left(\frac{1}{K}\right)^i \left(1-\frac{1}{K}\right)^{\ell-i} p^t(\underline{x})$$

where $x_i = y_{i+1}$ for $i = 1, 2, \ldots, R+K-1$

$$\lambda(m) = \begin{cases} 0 & m = 1 \\ m & m \neq 1 \end{cases}$$

and

$$\ell = \sum_{j=1}^{K} \lambda(x_{R+j})$$

Exchanging the order of the first and last summations, and evaluating their sum,

$$Q^{t+1}(\underline{z}) = \sum_{y_2=0}^{\infty} \cdots \sum_{y_{R+K}=0}^{\infty} \left(\prod_{j=2}^{R+K} z_j^{y_j} \right) \sum_{x_{R+K}=0}^{\infty} V^{t+1}(z_1) \left[1 - \frac{1}{K} + \frac{z_1}{K} \right]^{\ell} p^t(\underline{x})$$

Letting $y_i = x_{i-1}$ for i = 2, 3, ..., R+K and rearranging,

$$Q^{t+1}(\underline{z}) = V^{t+1}(z_1) \sum_{x_1=0}^{\infty} \cdots \sum_{x_{R+K}=0}^{\infty} \left(\prod_{j=1}^{R+K-1} z_{j+1}^{x_j} \right) \left[1 - \frac{1}{K} + \frac{z_1}{K} \right]^{\ell} p^t(\underline{x})$$

$$= V^{t+1}(z_1) \sum_{x_1=0}^{\infty} \cdots \sum_{x_{R+K}=0}^{\infty} \left(\prod_{j=1}^{R} z_{j+1}^{x_j} \right) \left(\prod_{j=R+1}^{R+K-1} z_{j+1}^{x_j} \left[1 - \frac{1}{K} + \frac{z_1}{K} \right]^{\lambda(x_j)} \right) \cdot$$

$$\left[1 - \frac{1}{K} + \frac{z_1}{K} \right]^{\lambda(x_{R+K})} \cdot p^t(\underline{x}) \qquad \text{(C.1)}$$

which is given by Eq. (4.3) and its accompanying algorithm.

Derivation of Eq. (4.4)

Define

$$h_i^{t-R-j} = \text{Prob[exactly } i \text{ packets retransmitting from the } (t-R-j)^{th} \text{ slot to the } t^{th} \text{ slot]}$$

We then have,

$$h_i^{t-R-j} = \begin{cases} P_0^{t-R-j} + P_1^{t-R-j} + \sum_{m=2}^{\infty} \left(1 - \frac{1}{K}\right)^m P_m^{t-R-j} & i = 0 \\ \sum_{m=2}^{\infty} \binom{m}{1} \left(\frac{1}{K}\right) \left(1 - \frac{1}{K}\right)^{m-1} P_m^{t-R-j} & i = 1 \\ \sum_{m=i}^{\infty} \binom{m}{i} \left(\frac{1}{K}\right)^i \left(1 - \frac{1}{K}\right)^{m-i} P_m^{t-R-j} & i \geq 2 \end{cases} \tag{C.2}$$

Now define

$$\hat{Q}^{t-R-j}(z) = \sum_{i=0}^{\infty} z^i h_i^{t-R-j}$$

Substituting Eqs. (C.2) into the above equation and summing, we get

$$\hat{Q}^{t-R-j}(z) = P_1^{t-R-j} \frac{(1-z)}{K} + \sum_{m=0}^{\infty} \left(1 - \frac{1}{K} + \frac{z}{K}\right)^m P_m^{t-R-j}$$

$$= P_1^{t-R-j} \frac{(1-z)}{K} + Q^{t-R-j} \left(1 - \frac{1}{K} + \frac{z}{K}\right) \tag{C.3}$$

Finally, by the weak independence assumption for channel traffic and the assumption that the channel input V^t is independent of the channel state,

$$Q^t(z) = V^t(z) \prod_{j=1}^{K} \hat{Q}^{t-R-j}(z) \tag{C.4}$$

which is the same as Eq. (4.1).

Theorem 4.1 and its proof

Theorem 4.1 If the channel input is an independent Poisson process, then the channel traffic is Poisson distributed in the limit as $K \to \infty$ under the weak independence assumption, such that

$$Q^t(z) = e^{-G^t(1-z)}$$

and

$$P_1{}^t = G^t e^{-G^t}$$

where

$$G^t = \frac{1}{K} \sum_{j=1}^{K} \left(G^{t-R-j} - G^{t-R-j} e^{-G^{t-R-j}} \right) + S^t$$

Proof Since V^t has a Poisson distribution,

$$V^t(z) = e^{-S^t(1-z)}$$

Substituting it into Eq. (4.4), we have

$$Q^t(z) = e^{-S^t(1-z)} \prod_{j=1}^{K} \left[Q^{t-R-j} \left(1 - \frac{1}{K} + \frac{z}{K} \right) + P_1{}^{t-R-j} \frac{1-z}{K} \right] \tag{C.5}$$

Consider

$$Q^{t-R-j} \left(1 - \frac{1}{K} + \frac{z}{K} \right) = P_0{}^{t-R-j} + \sum_{i=1}^{\infty} P_i{}^{t-R-j} \left[1 - \frac{i}{K}(1-z) \right] + o\left(\frac{1}{K}\right)$$

$$= 1 - G^{t-R-j} \frac{(1-z)}{K} + o\left(\frac{1}{K}\right) \tag{C.6}$$

where

$$\lim_{x \to 0} \frac{o(x)}{x} \to 0$$

Substituting Eq. (C.6) into Eq. (C.5) and letting $K \to \infty$

$$\lim_{K \to \infty} Q^t(z) = e^{-S^t(1-z)} \prod_{j=1}^{K} \left[1 - \left(G^{t-R-j} - P_1{}^{t-R-j} \right) \frac{(1-z)}{K} + o\left(\frac{1}{K}\right) \right]$$

$$= e^{-\left[\frac{1}{K} \sum_{j=1}^{K} \left(G^{t-R-j} - P_1{}^{t-R-j} \right) + S^t \right] (1-z)}$$

$$= e^{-G^t(1-z)} \tag{C.7}$$

where

$$G^t = \frac{1}{K} \sum_{j=1}^{K} \left(G^{t-R-j} - P_1{}^{t-R-j} \right) + S^t \tag{C.8}$$

From Eq. (C.7), we get

$$\lim_{K \to \infty} P_1{}^t = G^t \, e^{-G^t} \tag{C.9}$$

Q.E.D.

APPENDIX D

ALGORITHM 5.1, ITS DERIVATION AND

SOME MONOTONE PROPERTIES

Algorithm 5.1

This algorithm solves for the variables $\{t_i\}_{i=0}^{I}$ in the following set of $(I + 1)$ linear simultaneous equations,

$$t_0 = h_0 + \sum_{j=0}^{1} p_{0j}\, t_j \tag{D.1}$$

$$t_i = h_i + \sum_{j=i-1}^{I} p_{ij}\, t_j \qquad i = 1, 2, \ldots, I \tag{D.2}$$

(1) Define

$$e_I = 1$$

$$f_I = 0$$

$$e_{I-1} = \frac{1 - p_{II}}{p_{I,I-1}}$$

$$f_{I-1} = - \frac{h_I}{p_{I,I-1}}$$

(2) For $i = I - 1, I - 2, \ldots, 1$ solve recursively

$$e_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{I} p_{ij} e_j \right]$$

$$f_{i-1} = \frac{1}{p_{i,i-1}} \left[f_i - h_i - \sum_{j=i}^{I} p_{ij} f_j \right]$$

(3) Let

$$t_I = \frac{f_0 - h_0 - \sum_{j=0}^{I} p_{0j} f_j}{\sum_{j=0}^{I} p_{0j} e_j - e_0}$$

$$t_i = e_i t_I + f_i \qquad i = 0, 1, 2, \ldots, I - 1$$

Derivation of Algorithm 5.1

Define

$$t_i = e_i t_I + f_i \qquad i = 0, 1, 2, \ldots, I - 1 \qquad \text{(D.3)}$$

and

$$e_I = 1$$

$$f_I = 0 \qquad \text{(D.4)}$$

The last equation in Eqs. (D.2) is

$$t_I = h_I + p_{I,I-1}\, t_{I-1} + p_{II}\, t_I$$

Substituting $t_{i-1} = e_{I-1}\, t_i + f_{I-1}$ into the above equation, we get

$$t_I = h_I + p_{I,I-1}\, e_{I-1}\, t_I + p_{I,I-1}\, f_{I-1} + p_{II}\, t_I$$

Equating the coefficients of t_I and the constant terms, we have

$$e_{I-1} = \frac{1 - p_{II}}{p_{I,I-1}}$$

$$f_{I-1} = -\frac{h_I}{p_{I,I-1}} \qquad \text{(D.5)}$$

Eqs. (D.2) can be rewritten as follows,

$$t_{i-1} = \frac{1}{p_{i,i-1}} \left[t_i - h_i - \sum_{j=i}^{I} p_{ij}\, t_j \right] \qquad \text{(D.6)}$$

In each of the above equations, use Eqs. (D.3) to substitute for t_i. We then have

$$e_{i-1}\, t_I + f_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i\, t_I + f_i - h_i - \left(\sum_{j=i}^{I} p_{ij}\, e_j \right) t_I - \sum_{j=i}^{I} p_{ij}\, f_j \right]$$

Equating the coefficients of t_I and the constant terms, we get

$$e_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{I} p_{ij} \, e_j \right]$$

$$f_{i-1} = \frac{1}{p_{i,i-1}} \left[f_i - h_i - \sum_{j=i}^{I} p_{ij} \, f_j \right] \qquad \text{(D.7)}$$

From Eqs. (D.4), (D.5) and (D.7), e_i and f_i (i = I-2, I-3, ..., 1, 0) can then be determined recursively.

We next solve for t_I. Eqs. (D.3) are used to substitute for t_i in Eq. (D.1), which then becomes

$$e_0 \, t_I + f_0 = h_0 + \left(\sum_{j=0}^{I} p_{0j} \, e_j \right) t_I + \sum_{j=0}^{I} p_{0j} \, f_j$$

Solving for t_I in the above equation, we have

$$t_I = \frac{f_0 - h_0 - \sum_{j=0}^{I} p_{0j} \, f_j}{\sum_{j=0}^{I} p_{0j} \, e_j - e_0} \qquad \text{(D.8)}$$

Finally, t_i (i = 0, 1, 2, ..., I - 1) can be obtained from Eqs. (D.3), since e_i, f_i and t_I are all known. The derivation of Algorithm 5.1 is complete.

Some monotone properties

We show below monotone properties of the sequences e_i and f_i in Algorithm 5.1. The transition probabilities p_{ij} are assumed to be nonnegative and for each $i = 1, 2, \ldots$

$$\sum_{j=i-1}^{\infty} p_{ij} = 1$$

Also, the probabilities $p_{i,i-1}$ are assumed to be nonzero. (This last is a necessary condition for the Markov process in Section 5.1.1 to be irreducible.)

Property D.1 The sequence e_i is positive and monotonically decreases to one as i increases to I.

Proof From Eqs. (D.4) and (D.5),

$$e_I = 1$$

$$e_{I-1} = \frac{1 - p_{II}}{p_{I,I-1}} > 1$$

The proof is by induction. Assume that e_ℓ decreases as ℓ increases for $i \le \ell \le I$. From Eqs. (D.7)

$$e_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{I} p_{ij} e_j \right]$$

$$> \frac{e_i \left[1 - \sum_{j=i}^{I} p_{ij} \right]}{p_{i,i-1}} > e_i$$

Q.E.D.

Property D.2 (i) If $h_i > 0$, then the sequence f_i is negative and monotonically increases to zero as i increases to I. (ii) If $h_i < 0$, the sequence f_i is positive and monotonically decreases to zero as i increases to I.

Proof (i) From Eqs. (D.4) and (D.5),

$$f_I = 0$$

$$f_{I-1} = - \frac{h_I}{p_{I,I-1}}$$

The proof is by induction. Assume that f_ℓ increases as ℓ increases for $i \le \ell \le I$. From Eqs. (D.7)

$$f_{i-1} = \frac{1}{p_{i,i-1}} \left[f_i - h_i - \sum_{j=i}^{I} p_{ij} f_j \right]$$

$$< \frac{f_i \left[1 - \sum_{j=i}^{I} p_{ij} \right]}{p_{i,i-1}} < f_i$$

(ii) The proof is similar to that of (i).

Q.E.D.

APPENDIX E

ALGORITHM 6.5, ITS DERIVATION AND SOME MONOTONE PROPERTIES

Algorithm 6.5

This algorithm solves for g and $\{v_i\}_{i=1}^{M}$ in the following set of $(M + 1)$ linear simultaneous equations,

$$g = C_0 + \sum_{j=1}^{M} p_{0j} v_j \tag{E.1}$$

$$g + v_1 = C_1 + \sum_{j=1}^{M} p_{1j} v_j \tag{E.2}$$

$$g + v_i = C_i + \sum_{j=i-1}^{M} p_{ij} v_j \qquad i = 2, 3, \ldots, M \tag{E.3}$$

where

$$\sum_{j=0}^{M} p_{0j} = \sum_{j=i-1}^{M} p_{ij} = 1 \qquad i = 1, 2, \ldots, M \tag{E.4}$$

(1) Define

$$b_{M-1} = \frac{1}{p_{M,M-1}}$$

$$d_{M-1} = -\frac{C_M}{p_{M,M-1}}$$

(2) For $i = M - 1, M - 2, \ldots, 2$ solve recursively

$$b_{i-1} = \frac{1}{p_{i,i-1}} \left[b_i + 1 - \sum_{j=i}^{M-1} p_{ij} \, b_j \right]$$

$$d_{i-1} = \frac{1}{p_{i,i-1}} \left[d_i - C_i - \sum_{j=i}^{M-1} p_{ij} \, d_j \right]$$

(3) Define

$$u_M = - \frac{1}{p_{10}} \left[b_1 + 1 - \sum_{j=1}^{M-1} p_{1j} \, b_j \right]$$

$$w_M = - \frac{1}{p_{10}} \left[d_1 - C_1 - \sum_{j=1}^{M-1} p_{1j} \, d_j \right]$$

$$u_i = u_M + b_i \qquad i = 1, 2, \ldots, M - 1$$

$$w_i = w_M + d_i$$

(4) Let

$$g = \frac{C_0 + \sum_{j=1}^{M} p_{0j} \, w_j}{1 - \sum_{j=1}^{M} p_{0j} \, u_j}$$

$$v_i = u_i \, g + w_i \qquad i = 1, 2, \ldots, M$$

Derivation of Algorithm 6.5

Define

$$v_i = u_i\, g + w_i \qquad i = 1, 2, \ldots, M \tag{E.5}$$

The above equations are substituted into Eqs. (E.2) and (E.3) for the variables v_i. Equating the coefficients of g and the constant terms in the resulting equations, we obtain two sets of M linear simultaneous equations in terms of $\{u_i\}_{i=1}^{M}$ and $\{w_i\}_{i=1}^{M}$:

$$\begin{aligned} u_1 &= -1 + \sum_{j=1}^{M} p_{1j}\, u_j \\ u_i &= -1 + \sum_{j=i-1}^{M} p_{ij}\, u_j \qquad i = 2, 3, \ldots, M \end{aligned} \tag{E.6}$$

and

$$\begin{aligned} w_1 &= C_1 + \sum_{j=1}^{M} p_{1j}\, w_j \\ w_i &= C_i + \sum_{j=i-1}^{M} p_{ij}\, w_j \qquad i = 2, 3, \ldots, M \end{aligned} \tag{E.7}$$

Applying Algorithm 5.1 to Eqs. (E.6), we have

$$u_i = e_i \qquad i = 1, 2, \ldots, M - 1 \tag{E.8}$$

and

$$u_M = \frac{b_1 + 1 - \sum_{j=1}^{M} p_{1j}\, b_j}{\sum_{j=1}^{M} p_{1j}\, e_j - e_1} \tag{E.9}$$

where we define

$$\begin{aligned} e_M &= 1 \\ b_M &= 0 \end{aligned} \tag{E.10}$$

$$\begin{aligned} e_{M-1} &= \frac{1 - p_{MM}}{p_{M,M-1}} \\ b_{M-1} &= \frac{1}{p_{M,M-1}} \end{aligned} \tag{E.11}$$

and for $i = M - 1, M - 2, \ldots, 2$ we solve recursively

$$\begin{aligned} e_{i-1} &= \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{M} p_{ij}\, e_j \right] \\ b_{i-1} &= \frac{1}{p_{i,i-1}} \left[b_i + i - \sum_{j=i}^{M} p_{ij}\, b_j \right] \end{aligned} \tag{E.12}$$

Similarly, applying Algorithm 5.1 to Eqs. (E.7), we have

$$w_i = f_i\, w_M + d_i \qquad i = 1, 2, \ldots, M - 1 \tag{E.13}$$

and

$$w_M = \frac{d_1 - C_1 - \sum_{j=1}^{M} p_{1j}\, d_j}{\sum_{j=1}^{M} p_{1j}\, f_j - f_1} \tag{E.14}$$

where we define

$$\begin{aligned} f_M &= 1 \\ d_M &= 0 \end{aligned} \tag{E.15}$$

$$\begin{aligned} f_{M-1} &= \frac{1 - p_{MM}}{p_{M,M-1}} \\ d_{M-1} &= -\frac{C_M}{p_{M,M-1}} \end{aligned} \tag{E.16}$$

and for $i = M - 1, M - 2, \ldots, 2$ we solve recursively

$$\begin{aligned} f_{i-1} &= \frac{1}{p_{i,i-1}} \left[f_i - \sum_{j=i}^{M} p_{ij}\, f_j \right] \\ d_{i-1} &= \frac{1}{p_{i,i-1}} \left[d_i - C_i - \sum_{j=i}^{M} p_{ij}\, d_j \right] \end{aligned} \tag{E.17}$$

We note from Eqs. (E.10)-(E.12) and Eqs. (E.15)-(E.17) that $e_i = f_i$ for $i = 1, 2, \ldots, M$. We proceed to show that $e_i = f_i = 1$ for all i. From Eqs. (E.10) and (E.11)

$$e_M = 1$$

$$e_{M-1} = \frac{1 - p_{MM}}{p_{M,M-1}} = 1$$

This last is true by virtue of Eqs. (E.4). We now use induction and assume that

$$e_\ell = 1 \qquad \ell = M, M - 1, \ldots, i$$

From Eqs. (E.12),

$$e_{i-1} = \frac{1}{p_{i,i-1}} \left[e_i - \sum_{j=i}^{M} p_{ij} e_j \right]$$

$$= \frac{1}{p_{i,i-1}} \left(1 - \sum_{j=i}^{M} p_{ij} \right) = 1$$

Thus, by induction we have shown that $e_i = f_i = 1$ for all i.

Using the preceding result, the solution to the set of M linear simultaneous equations in Eqs. (E.6) now becomes,

$$u_i = u_M + b_i \qquad i = 1, 2, \ldots, M - 1 \qquad \text{(E.18)}$$

and

$$u_M = -\frac{1}{p_{10}}\left(b_1 + 1 - \sum_{j=1}^{M-1} p_{1j}\, b_j\right) \tag{E.19}$$

where we define

$$b_{M-1} = \frac{1}{p_{M,M-1}} \tag{E.20}$$

and for $i = M - 1, M - 2, \ldots, 2$ we obtain recursively

$$b_{i-1} = \frac{1}{p_{i,i-1}}\left(b_i + 1 - \sum_{j=i}^{M-1} p_{ij}\, b_j\right) \tag{E.21}$$

Similarly, the solution to the set of M linear simultaneous equations in Eqs. (E.7) becomes

$$w_i = w_M + d_i \qquad i = 1, 2, \ldots, M - 1 \tag{E.22}$$

and

$$w_M = -\frac{1}{p_{10}}\left(d_i - C_1 - \sum_{j=1}^{M-1} p_{1j}\, d_j\right) \tag{E.23}$$

where we define

$$d_{M-1} = -\frac{C_M}{p_{M,M-1}} \tag{E.24}$$

and for $i = M - 1, M - 2, \ldots, 2$ we obtain recursively

$$d_{i-1} = \frac{1}{p_{i,i-1}} \left(d_i - C_i - \sum_{j=i}^{M-1} p_{ij}\, d_j \right) \qquad \text{(E.25)}$$

Using Eqs. (E.5) to substitute for v_i in Eq. (E.1), we obtain

$$g = C_0 + \left(\sum_{j=1}^{M} p_{0j}\, u_j \right) g + \sum_{j=1}^{M} p_{0j}\, w_j$$

from which we get

$$g = \frac{C_0 + \sum_{j=1}^{M} p_{0j}\, w_j}{1 - \sum_{j=1}^{M} p_{0j}\, u_j} \qquad \text{(E.26)}$$

Finally, v_i are obtained using Eqs. (E.5). The derivation of Algorithm 6.5 is complete.

Some monotone properties

We show below monotone properties of the sequences b_i, d_i, u_i and w_i in Algorithm 6.5. The transition probabilities p_{ij} are assumed to satisfy Eqs. (E.4). The probabilities $p_{i,i-1}$ are assumed to be strictly positive. (This last is a necessary condition for the Markov process in Section 6.3 to be irreducible.)

Property E.1 The sequence b_i is positive and monotonically decreases to 0 as i increases to M.

Proof From Eqs. (E.10) and (E.11),

$$b_M = 0$$

$$b_{M-1} = \frac{1}{p_{M,M-1}} > b_M$$

The proof is by induction. Assume that b_ℓ decreases as ℓ increases for $i \le \ell \le M$. From Eqs. (E.12) and (E.4)

$$b_{i-1} = \frac{1}{p_{i,i-1}} \left[b_i + 1 - \sum_{j=i}^{M} p_{ij} b_j \right]$$

$$> \frac{1 + b_i p_{i,i-1}}{p_{i,i-1}} > b_i$$

Q.E.D.

Property E.2 (i) If C_i are positive, the sequence $_i$ is negative and monotonically increases to 0 as i increases to M. (ii) If C_1 are negative, the sequence d_i is positive and monotonically decreases to 0 as i increases to M.

Proof The proof uses Eqs. (E.4), (E.15), (E.16) and (E.17), and is similar to that of Property E.1.

Property E.3 The sequence u_i is negative and monotonically decreases as i increases.

Proof From Eq. (E.19)

$$u_M = -\frac{1}{p_{10}}\left(b_1 + 1 - \sum_{j=1}^{M-1} p_{1j}\, b_j\right)$$

$$-u_M = \frac{1}{p_{10}}\left(1 + b_1 - \sum_{j=1}^{M-1} p_{1j}\, b_j\right)$$

$$> \frac{1 + b_1\, p_{10}}{p_{1j}} > b_1$$

where b_1 is positive from Property E.1. From Eq. (E.18)

$$u_j = u_N + b_j$$

Applying Property E.1, the proof is complete.

Q.E.D.

Property E.4 (i) If C_i are positive, the sequence w_i is positive and monotonically increases as i increases. (ii) If C_i are negative, the sequence w_i is negative and monotonically decreases as i increases.

Proof The proof uses Eqs. (E.22) and (E.23) and Property E.2. The proof is similar to that of Property E 3.

APPENDIX F

A GENERAL DYNAMIC CHANNEL CONTROL PROCEDURE

In this appendix, a dynamic channel control procedure is formulated which includes as special cases ICP, RCP and IRCP in Chapter 6. Lemma 6.3 and Theorem 6.4 on the equivalence of the performance measures for ICP, RCP and IRCP are then extended to this general case.

Consider the action space $A_1 = \{\beta_1, \beta_2, \ldots, \beta_m\}$ where $0 \le \beta_1 < \beta_2 < \ldots < \beta_m \le 1$, and the action space $A_2 = \{\gamma_1, \gamma_2, \ldots, \gamma_k\}$ where $0 < \gamma_1 < \gamma_2 < \ldots < \gamma_k < 1$. Let $A = A_1 \times A_2$ such that each element in A is a two-dimensional vector (β, γ). As in Section 6.3, the Markov decision process N^t has a finite state space $S = \{0, 1, 2, \ldots, M\}$. A stationary control policy f maps S into A. Given a stationary control policy f, $f(i) = (\beta, \gamma)$ means that whenever $N^t = i$, each (new) packet arrival is accepted with probability β (and rejected with probability $1 - \beta$) while each backlogged packet is retransmitted with probability γ in the t^{th} time slot. Thus, ICP corresponds to the special case $A = \{0, 1\} \times \{p_o\}$; RCP corresponds to the special case $A = \{1\} \times \{p_o, p_c\}$; IRCP corresponds to the special case $A = \{0, 1\} \times \{p_o, p_c\}$.

State Transition Probabilities

Suppose N^t is in state i and the stationary control policy $f(i) = (\beta, \gamma)$, then the one-step state transition probabilities are given by

$$p_{ij}(f) = \begin{cases} 0 & j \le i - 2 \\ i\gamma(1-\gamma)^{i-1}(1-\beta\sigma)^{M-i} & j = i - 1 \\ (1-\gamma)^{i}(M-i)\beta\sigma(1-\beta\sigma)^{M-i-1} & \\ \quad + \left[1 - i\gamma(1-\gamma)^{i-1}\right](1-\beta\sigma)^{M-i} & j = i \\ \left[1 - (1-\gamma)^{i}\right](M-i)\beta\sigma(1-\beta\sigma)^{M-i-1} & j = i+1 \\ \binom{M-i}{j-i}(\beta\sigma)^{j-i}(1-\beta\sigma)^{M-j} & j \ge i + 2 \end{cases}$$

$$0 \le i, j \le M \qquad \text{(F.1)}$$

Stationary Channel Throughput Rate

Suppose N^t is in state i and $f(i) = (\beta, \gamma)$. Define the expected immediate cost to be

$$C_i(f) = -S_{out}(i, f)$$

$$= -\left[i\gamma(1-\gamma)^{i-1}(1-\beta\sigma)^{M-i} + (1-\gamma)^{i}(M-i)\beta\sigma(1-\beta\sigma)^{M-i-1}\right] \qquad \text{(F.2)}$$

By Eq. (6.9) the cost rate of N^t is

$$g_s(f) = -\sum_{i=0}^{M} \pi_i(f)\, S_{out}(i, f)$$

Then, the stationary channel throughput rate is given by Eq. (6.30) which we rewrite below.

$$S_{out} = -g_s(f) \qquad \text{(F.3)}$$

Average Packet Delay

Suppose N^t is in state i and $f(i) = (\beta, \gamma)$. Define the expected immediate cost to be

$$C_i(f) = i + (M - i)(1 - \beta)\sigma d_r \tag{F.4}$$

where d_r is the expected cost in units of delay per packet arrival rejected and is equal to $\frac{1}{\sigma}$ (see Section 6.3.3).

Let $S = \bigcup_{\ell=1}^{m} S_\ell$ where $S_1, S_2, \ldots, S_m$ are nonintersecting sets corresponding to a stationary control policy f such that

$$f(i) = (\beta_\ell, \gamma) \quad \text{if and only if} \quad i \in S_\ell$$

where $\ell = 1, 2, \ldots, m$ and γ is any action in A_2.

By Eq. (6.9), the cost rate of N^t is

$$g_d(f) = \sum_{i=0}^{M} C_i(f)\pi_i(f)$$

$$= \sum_{i=0}^{M} i\,\pi_i(f) + \sum_{\ell=1}^{m} \sum_{i \in S_\ell} (M - i)(1 - \beta_\ell)\sigma d_r \pi_i(f)$$

$$= \bar{N} + \lambda_r d_r$$

$$= \bar{N} + \bar{N}_r \tag{F.5}$$

where

$$\lambda_r = \sum_{\ell=1}^{m} \sum_{i \in S_\ell} (M - i)(1 - \beta_\ell)\sigma\, \pi_i(f) \tag{F.6}$$

is the stationary packet rejection rate; $\bar{N}$ is the average channel backlog size and $\bar{N}_r$ is the average number of rejected packets in the system.

Using Little's result [LITT 61] the average packet delay is given by Eq. (6.31) which we rewrite below.

$$D = \frac{g_d(f)}{S_{out}} + R + 1$$

$$= - \frac{g_d(f)}{g_s(f)} + R + 1 \qquad (F.7)$$

We give below an extension of Lemma 6.3 to the general dynamic channel control procedure.

Lemma F.1 Given any stationary control policy $f: S \to A$

$$g_d(f) = \frac{g_s(f)}{\sigma} + M$$

Proof From Eq. (F.5) and $d_r = \frac{1}{\sigma}$

$$g_d(f) = \sum_{i=0}^{M} i\, \pi_i(f) + \sum_{i=0}^{M} (M - i)\, \pi_i(f)$$

$$- \frac{1}{\sigma} \sum_{\ell=1}^{m} \sum_{i \in S_\ell} (M - i)\, \beta_\ell \sigma\, \pi_i(f)$$

$$= M - \frac{1}{\sigma} \sum_{\ell=1}^{m} \sum_{i \in S_\ell} (M - i) \beta_\ell\, \sigma\, \pi_i(f)$$

Note that $\sum_{\ell=1}^{m} \sum_{i \in S_\ell} (M - i)\beta_\ell\, \sigma\, \pi_i(f)$ is just the stationary channel input rate and is thus equal to the stationary channel throughput rate $S_{out} = - g_s(f)$. Hence,

$$g_d(f) = \frac{g_s(f)}{\sigma} + M$$

Q.E.D.

With the above lemma, Theorem 6.4 can then be extended to the general dynamic channel control procedure.